Dankwoord

Graag wil ik iedereen bedanken die heeft bijgedragen tot het tot stand komen van dit doctoraatswerk.

Mijn dank gaat in de eerste plaats naar mijn promotor, Professor J. De Schutter, voor zijn steun en daadwerkelijke begeleiding. Zijn suggesties en stimulatie maakten het succesvol voltooien van dit onderzoekswerk mogelijk. Zijn enthousiasme en inzet voor lesgeven en onderzoek, zullen voor mij een inspirerend voorbeeld blijven.

Een bijzondere waardering heb ik voor Professor H. Van Brussel aan wie ik het voorrecht dank te hebben mogen werken aan de afdeling Produktietechnieken, Machinebouw en Automatisering. Eveneens dank ik hem voor de betoonde interesse voor mijn onderzoek en voor het zorgvuldig nalezen van dit manuscript.

Ook Professor J. Vandewalle wil ik bedanken voor de kostbare tijd die hij besteedde aan het nalezen van dit manuscript.

The work presented in this thesis is carried out in the framework of ESPRIT Project 1561 ("SACODY") with partners: Bertin & Cie (F), Katholieke Universiteit Leuven (B), AEG AG (D), Leuven Measurements and Systems (B), KUKA Schweissanlagen und Robotor GmbH (D), and University College Dublin (IR). I would like to thank all partners for their support throughout the project. I would like to acknowledge the support of the Commission of the European Communities, as represented by DG XIII, on Telecommunications, Informations Industries and Innovation, and more especially its CIM Group.

I would like to thank J.L. Faillot and A. Froment of Bertin & Cie

for their cooperation in some tasks of the ESPRIT project. It has to be mentioned that the first formulation of the nonlinear control method, used in this dissertation, is made by them.

Ik wens ook alle SACODY-medewerkers van de afdeling PMA te bedanken: Hans Thielemans voor het beheer van het VME-systeem, Francky Demeester voor zijn hulp en raadgevingen, Christophe Jankowski voor zijn hulp bij het modeleren, Johan Van Den Bossche voor "RODYM", en Luk Persoons en Jan Leroy voor het ontwerp en de realisatie van de testopstellingen. Een speciaal woord van dank gaat naar Jan Swevers en Dirk Torfs, voor de samenwerking aan de vele deeltaken van het project, de mogelijkheid tot discussie en de praktische tips en raadgevingen die vooral de laatste loodjes aanzienlijk hebben verlicht.

Verder zou ik Paul Vanherck oprecht willen bedanken voor zijn aanmoedigingen en practische hulp en voor zijn inzicht in, en zijn aanvaarden van de beperkingen voorkomend bij praktische realisaties.

Dankbaar denk ik terug aan de jaren doorgebracht aan de afdeling PMA. Hierbij dank ik mijn collega's, het personeel van de electronische en mechanische werkplaats en de secretaressen. Een speciaal woord van dank gaat naar mijn opeenvolgende bureaugenoten voor de aangename sfeer en voor de geboden hulp. Philippe Vanherck dank ik voor het steeds klaarstaan voor het oplossen van problemen met het VAX-systeem. Ook dank ik de opeenvolgende eindwerkstudenten voor het meewerken aan de practische realisaties.

Tenslotte wil ik Dirk bedanken voor zijn vertrouwen en voor zijn aanmoedigingen om dit werk te realiseren en te voltooien.

Marleen Adams
Heverlee, februari 1992

system design is the knowledge of this model. To this end, modeling of flexible manipulators and identification of these models is reviewed. When implementing control algorithms on experimental robots, the problem of nonlinear motor friction has to be dealt with. To this end, an overview in techniques to minimize the influence of nonlinear motor friction on the controller, is given.

1.2.1 Control of rigid manipulators

Manipulators are multi–input, multi–output systems, described by highly nonlinear equations with nonlinear couplings between the variables of motion. However, the present commercial control systems make use of decoupled control [18], [48]. Strong simplifications are made in the system description; cross inertias and coriolis and centrifugal forces are neglected so that the multi–input, multi–output system appears as a decoupled set of single–input, single–output systems. In addition the inertia and gravitation forces are assumed to be position independent, so that classical linear servo control can be applied. Because of these strong simplifications, the dynamic behaviour of the decoupled control systems is not optimal.

The use of decoupled control systems in industry has several reasons [48]. The knowledge of a dynamic robot model, which has to be identified by means of experimental identification procedures, is not required. The implementation of these decoupled control systems relies on rather straightforward rules. Hence, it can be performed easily. In addition the hardware requirements, concerning processing speed, sampling rate and memory volume are low.

The advanced manufacturing problems of the next years will demand increased performance, especially concerning velocity and dynamic accuracy, which can no longer be met by the present commercial control systems. In the last 15 years a large number of control algorithms for rigid robots has been developed in the robot control research area [19], [48].

The most advanced controllers are based on nonlinear decoupling (computed torque methods). By compensation of the nonlinear terms in the rigid body dynamics, the system is reduced to a set of linear decoupled subsystems (1 subsystem per link). Each of these subsystems is then controlled by a classical decoupled linear feedback loop given

Abstract

Flexibility of joints or links limits the performance of robots. Traditional control designed for rigid robots must be supplemented. First, to prevent or damp out undesired oscillations resulting from excitation of structural dynamics and second, to improve the tracking of a reference trajectory designed for a rigid robot.

The model describing the dynamic behaviour of multi-link flexible robots is characterised by highly nonlinear equations with nonlinear couplings between the variables of motion. This thesis proposes a nonlinear control method applicable to multi–link robots with flexible links and/or flexible joints. The method consists of three steps:
(1) a nonlinear feedforward to cancel out nonlinear terms in the rigid body dynamics;
(2) linearization of the remaining nonlinearities, and linear feedback of all state variables (rigid and flexible coordinates). This to ensure proper stability of the system about the desired state;
(3) compensation of static and dynamic deflection to improve the tracking of the reference trajectory. This compensation is achieved by calculation of the reference values of all state variables based on the model of the flexible robot.

The control algorithm is verified experimentally, first on test set ups, then on an industrial robot. Special attention is given to the identification of the nonlinear models describing the dynamic behaviour of the systems, to state estimation, to Coulomb friction compensation and to trajectory generation.

Contents

Nomenclature

The most important notations are summarized below. Symbols that occur only locally in one or a few sections are not mentioned here. All symbols used, including the ones below, are explained in the text. Bold lower case letters (e.g. $\mathbf{v}$) are used for vectors, while bold capital letters (e.g. $\mathbf{M}$) are used for matrices.

General symbols and conventions

$\mathbf{I}$	: identity matrix,
j	: $\sqrt{-1}$,
k	: discrete time step,
s	: Laplace transform variable,
t	: time,
T_s	: sampling period,
f_s	: sample frequency,
z	: z-transform variable,
$\mathbf{0}$	: zero matrix,
ω	: pulsation, frequency,
ξ	: damping ratio,
ω_z	: anti–resonance frequency,
ω_p	: resonance frequency,
$\Re(\bullet)$	: real part of a complex number,
$\Im(\bullet)$	: imaginary part of a complex number,
$\bullet_d$	: desired value of variable or vector,
$\bullet_f$	: filtered variable,
$\bullet^{-1}$	: inverse of a matrix,
$\bullet^T$	: transpose of a matrix,
$\hat{\bullet}$	: estimated value,

$\dot{\bullet}$	: first derivative with respect to time or velocity,
$\ddot{\bullet}$	: second derivative with respect to time or acceleration,
$\bullet^{III}$	: third derivative with respect to time or jerk,
$\bullet^{IV}$	: fourth derivative with respect to time,
$\frac{\delta F}{\delta x_i}$	: partial derivative of funtion F with respect to the variable x_i,
$\frac{\delta \mathbf{F}}{\delta \mathbf{x}}\|_{\mathbf{x}}$	: vector of partial derivatives of vector function $\mathbf{F}$ with respect to the elements of vector $\mathbf{x}$ (jacobian matrix),
$\|\bullet\|$	: absolute value,
$sgn(v)$	: sign function, = 1 if $v > 0$, = 0 if $v = 0$, = -1 if $v < 0$,
$\bullet_i$, $\bullet_j$, $\bullet_k$, $\bullet_r$	: current indices.

Modeling

O_i	: ith joint between link $i-1$ and link i,
O_i'	: endpoint of link i,
$O_i x_i y_i$	: body fixed moving coordinate system i, the axis $O_i x_i$ is directed along the tangent of the center line of link i in O_i,
$O_i' x_i' y_i'$	: body fixed moving coordinate system i, the axis $O_i' x_i'$ is directed along the tangent of the center line of link i in O_i',
$O_1 XY$	: inertial coordinate frame,
$\mathbf{C_i}$	: transformation matrix between coordinate system $O_i x_i y_i$ and $O_1 XY$,
$\theta_i = \theta_{li}$	: joint displacement measured with respect to the coordinate axis $O_{i-1}' x_{i-1}'$,
θ_i'	: angle between coordinate axis $O_{i-1} x_{i-1}$ and $O_i x_i$,
$v_i(x_i, t)$	: lateral deformation of link beam i,
n_i	: number of modes used to describe the deflection of link i,
$\Phi_{i,r}(x_i)$	: mode shape functions of link beam i, $r = 1, ..., n_i$,

$q_{l\,i,r}(t)$: time dependent modal amplitudes or weights of link beam i, $r = 1, ..., n_i$,

$O_i x_i'' y_i''$, : body fixed moving coordinate system i, defined so that the mode shape functions and the rigid body mode x_i'', displacement along coordinate axis x_i'', are orthogonal functions,

$\Psi_{i,r}(x_i'')$: orthogonal mode shape functions of link beam i, $r = 1, ..., n_i$ in body fixed moving coordinate system $O_i x_i'' y_i''$,

α_i : angle between coordinate axis $O'_{i-1} x'_{i-1}$ and $O_i x_i''$,

$\mathbf{Q}$: generalized force vector,

E_K : kinetic energy,

E_P : potential energy,

E_{Pe} : elastic potential energy,

E_{Pg} : gravitational potential energy,

N : number of rigid degrees of freedom,

P : number of flexible degrees of freedom,

$\mathbf{T}$: vector of motor torques (elements T_i),

$\boldsymbol{\theta}_m$: vector of angular motor positions or joint angles before the elastic transmissions (elements θ_{mi}),

$\boldsymbol{\theta}_l$: vector of angular link positions or joint angles after the elastic transmissions (elements θ_{li}),

$\boldsymbol{\theta}_r$: vector of angular positions of virtual rigid links,

r_i : transmission gear ratio of joint i,

$\mathbf{q}_0$: vector of rigid body coordinates, equal to $\boldsymbol{\theta}_m$ for flexible joint robots and equal to $\boldsymbol{\theta}_l = \boldsymbol{\theta}_m = \boldsymbol{\theta}$ for rigid joint robots,

$\mathbf{q}_l$: vector of generalized coordinates associated with the link flexibility (elements $q_{l\,i,r}$, $i = 1$, .., N, $r = 1, ..., n_i$),

$\mathbf{q}_j$: vector of generalized coordinates associated with the joint flexibility (the elements q_{ji} are equal to $\theta_{li} - \frac{\theta_{mi}}{r_i}$,),

if there is no possibility of confusion, the subscript l or j of $\mathbf{q}$ is omitted.

K_{ji} : spring constant describing the flexibility of joint i,

$\mathbf{q}_f$: vector of flexible coordinates, equal to $[\mathbf{q}_j \quad \mathbf{q}_l]^T$,

$y_i(t)$: tip rotation of the ith link,

E_i : Young modulus of the ith link,

$(EI)_i$: bending stiffness of the ith link,

ρ_i : density of the ith link,

S_i : cross section area of the ith link,

l_i : lenght of the ith link (distances between joints),

m_i : ith link mass,

x_{ic} : ith center of mass position,

m : mass,

I : moment of inertia,

$\bullet_t$: total, with respect to axis,

$\bullet_l$: of link,

$\bullet_p$: of payload,

$\bullet_h$: of hub or joint,

$\bullet_b$: of beam,

$\bullet_z$: with respect to axis,

I_{mi} : the rotor/gear assembly inertia of joint i,

T_{fi} : total motor friction torque of motor i,

F_{mi} : Coulomb friction coefficient of motor i,

C_{mi} : linear viscous friction coefficient of motor i,

$\omega_{i,j}$: jth cantilevered mode eigenpulsation of the ith link,

$\zeta_{i,j}$: modal damping of the jth mode of the ith link,

$\eta_{i,j}$: $\int_0^{l_i} \rho_i S_i \Phi_{i,j}(x_i) dx_i$,

$K_{i,j}$: $\int_0^{l_i} \rho_i S_i x_i \Phi_{i,j}(x_i) dx_i$,

$m_{i,j}$: $\int_0^{l_i} \rho_i S_i \Phi_{i,j}^2(x_i) dx_i$ generalized mass of jth mode of link i,

$m_{i,j,k}$: $\int_0^{l_i} \rho_i S_i \Phi_{i,j}(x_1) \Phi_{i,k}(x_1) dx_1$,

$k_{i,j}$: $\int_o^{l_i} (EI)_i \left[\frac{\delta^2 \Phi_{i,j}}{\delta x_i^2}\right]^2 dx_i = m_{i,j}\omega_{i,j}^2$,

$\mathbf{M}(\boldsymbol{\theta}_l)$: generalized inertia matrix of a flexible manipulator,

$\mathbf{W}(\boldsymbol{\theta}_l)$: inverse of the generalized mass matrix of a flexible manipulator,

$\mathbf{M}_r(\boldsymbol{\theta}_l) = \mathbf{M}_{11}(\boldsymbol{\theta}_l)$: inertia matrix of a rigid manipulator,

$\mathbf{C}(\theta_l, \mathbf{q}_l, \dot{\theta}_l, \dot{\mathbf{q}}_l)$: the nonlinear centrifugal and coriolus terms vector,
$\mathbf{G}(\theta_l, \mathbf{q}_l)$: gravity vector,
$\mathbf{K}_{fl} = Diag\{k_{i,j}\}$: the modal generalized stiffness matrix,
$\mathbf{D}_{fl} = Diag\{d_{i,j}\}$: the structural damping matrix,
$\mathbf{K}_j = Diag\{K_{ji}\}$: matrix containing the spring constants of the joints,
$\mathbf{J} = Diag\{I_{mi}\}$: rotor/gear assembly inertia matrix,
$\mathbf{F}_r(\theta_l, \dot{\theta}_l)$: vector containing centrifugal, coriolus and gravitational forces for a rigid manipulator,
$\mathbf{F}_{\mathbf{q}_0}(\mathbf{q}_0, \mathbf{q}_f, \dot{\mathbf{q}}_0, \dot{\mathbf{q}}_f)$: vector containing centrifugal, coriolis, gravitational and modal stiffness forces for a flexible manipulator (associated with the rigid body dynamics),
$\mathbf{F}_{\mathbf{q}_f}(\mathbf{q}_0, \mathbf{q}_f, \dot{\mathbf{q}}_0, \dot{\mathbf{q}}_f)$: vector containing centrifugal, coriolus, gravitational and modal stiffness forces for a flexible manipulator (associated with the flexible dynamics).

For one–link robots the subscript i referring to the ith link, motor or joint is omitted.

Linear identification

$U(z^{-1})$: z-transform of the input,
$Y(z^{-1})$: z-transform of the output,
$H(z^{-1})$: discrete transfer function to be identified,
$A(z^{-1})$: denominator polynomial of $H(z^{-1})$,
$B(z^{-1})$: numerator polynomial of $H(z^{-1})$,
n_a : order of the denominator of the discrete transfer function,
n_b : order of the numerator of the discrete transfer function,
a_i : denominator coefficient, $i = 0, .., n_a$,
b_i : numerator coefficient, $i = 0, .., n_b$,
θ_0 : parameter vector containing the numerator and denominator parameters.

Control

$\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$	: continuous time state space model matrices,
$\mathbf{F}$, $\mathbf{G}$, $\mathbf{H}$, $\mathbf{J}$	: discrete time state space model matrices,
$\mathbf{x}$	: state vector,
$\mathbf{x}_d$	: desired state vector,
$\Delta\mathbf{x}$	: error $\mathbf{x} - \mathbf{x}_d$,
$\mathbf{y}$	: output vector,
$\mathbf{K}$	: feedback gain matrix (elements K_i),
$\mathbf{K}_p$	: diagonal matrix consisting of position feedback gains,
$\mathbf{K}_v$	: diagonal matrix consisting of velocity feedback gains,
k_i	: integral feedback gain,
$\mathbf{T}_{ff}$	: vector of feedforward torques,
$\mathbf{T}_{fb}$	: vector of feedback torques,
$\mathbf{T}_{int}$	: vector of integration inputs,
$\mathbf{T}_c$	: vector of Coulomb friction canceling torques,
$\mathbf{T}_t$	: vector of total control torques,
J	: quadratic performance index,
$\mathbf{Q}$, $\mathbf{R}$	: positive–definite real symmetrix weighting matrices,
T_s	: sampling period,
$\mathbf{L}$	: feedback estimator gain matrix,
$\hat{\bullet}$	: estimated value; measurement update,
$\tilde{\bullet}$	: estimated value; time update.

Experimental flexible one-link robot

u	: input command to the motor (in $Volt$),
ε	: strain gauge signal (in $\mu m/m$),
k_ε	: proportional constant between flexible coordinate q and ε,
k_u	: proportional constant between input command u and motor torque,
$H_1(\bullet)$	: transfer function relating u to ε, $\bullet = s$, or z^{-1},
$H_2(\bullet)$	: transfer function relating ε to θ, $\bullet = s$, or z^{-1},

$H_3(\bullet)$: transfer function relating u to θ, $\bullet = s$, or z^{-1}.

Nederlandse samenvatting

Inleiding en motivatie

Industriële robots zullen in de toekomst moeten voldoen én aan een verhoogde nauwkeurigheid én aan steeds hogere vereisten betreffende snelheid, versnelling en werkingsgebied.

Structurele flexibliteit vormt een beperking voor de prestaties van de robot. Flexibiliteit wordt veroorzaakt door elasticiteit van één of meerdere geledingen of door elasticiteiten optredend in de gewrichten of door een combinatie van beide.

Traditionele robots worden zo stijf mogelijk gebouwd om voldoende nauwkeurigheid te bekomen. Zelfs dan nog blijft de flexibiliteit van de aandrijvingen een belangrijk probleem. De stijfheid en het gewicht van deze traditionele robots beperkt de bereikbare snelheid en versnelling. Daarom moeten gewicht en inertie van bewegende delen verminderd worden. Dit leidt tot lagere structurele resonantiefrekwenties.

Flexibiliteit beperkt de bereikbare bandbreedte van traditionele besturingen die ontworpen zijn voor stijve robots. De prestaties van deze besturingen verslechteren zeer snel wanneer de bandbreedte van de gesloten kring de laagste structurele resonantiefrekwentie bereikt. Een vuistregel in de besturing van robots is dat de bandbreedte moet beperkt worden tot minder dan de helft van de laagste resonantiefrekwentie. Het gevolg is dat structurele oscillaties onvoldoende worden gedempt door de besturing. Die oscillaties treden op wanneer de systeemresonanties opgewekt worden door de robot te bruusk te bewegen. Dit is de reden waarom flexibiliteit de efficiëntie van puntbesturingen beperkt. Alvorens een operatie te starten in het punt van bestemming is het noodzakelijk te wachten tot trillingen uitgestorven zijn. Flexibiliteit heeft eveneens statische doorbuiging door de zwaartekracht tot gevolg. Dit veroorzaakt positiefouten tussen de werkelijke positie van

het roboteindpunt en het gewenste instelpunt.

Bij contourbesturingen, waarvan het doel is het roboteindpunt een gewenst traject zo nauwkeurig mogelijk te laten volgen, beperkt de flexibiliteit de volgnauwkeurigheid. De volgfouten worden veroorzaakt door doorbuigingen ten gevolge van inertie, coriolis en centripetaalkrachten. Bijvoorbeeld om van een punt naar een ander punt te bewegen, met begin- en eindsnelheid gelijk aan nul, moet de flexibele robot versnellen en vertragen. Dit veroorzaakt inertiekrachten die dan resulteren in overeenkomstige doorbuigingen.

Het doel van deze verhandeling is de traditionele besturingen aan te vullen ten eerste om ongewenste trillingen te voorkomen of te dempen en ten tweede om het volgen van een referentietrajekt te optimaliseren. Daarvoor werd een besturingsalgotritme ontwikkeld en experimenteel geverifiëerd eerst op testopstellingen en daarna op een industriële robot.

Deze verhandeling bestaat uit twee delen. In een eerste deel wordt het ontwerp van een niet–lineaire besturing voor flexibele robots beschreven. Het tweede deel geeft simulatie en experimentele resultaten. De volgende secties vatten de belangrijkste resultaten van elk hoofdstuk samen.

Deel 1: Niet–lineaire besturing van flexibele robots

Hoofdstuk 1: Literatuur overzicht

Besturing van flexibele robots met één gelid moet onderscheiden worden van besturing van flexibele robots met verscheidene geledingen. Het dynamisch model van een robot met één flexibel gelid is lineair, klassieke lineaire identifikatie- en besturingstechnieken kunnen dus gebruikt worden.

Een flexibele robot met verscheidene geledingen is een systeem met meerdere ingangen en uitgangen, beschreven door een stelsel niet–lineaire en gekoppelde differentiaalvergelijkingen.

In het midden van de jaren 1970 ontwikkelde Freund [28] de niet–lineaire ontkoppelingstheorie. Uitgaande van het dynamisch model van de robot, wordt het koppel berekend dat moet geleverd worden aan elke motor, zodanig dat het niet–lineaire systeem met meerdere ingangen

en uitgangen gereduceerd wordt tot een set van lineaire ontkoppelde deelsystemen (1 deelsysteem per gelid). Voor elk deelsysteem kan dan een eenvoudige lineaire besturing ontworpen worden. In de literatuur wordt deze methode ook aangeduid met "computed torque", niet–lineaire toestandsterugkoppeling of inverse dynamica.

De niet–lineaire ontkoppelingstheorie werd ontwikkeld voor robots met stijve geledingen. Deze klasse van systemen zijn linearizeerbaar door toepassing van niet–lineaire toestandsterugkoppeling.

Wanneer de flexibiliteit van de geledingen in rekening gebracht wordt, kan echter bewezen worden [74] dat linearizatie van zowel de beweging van het starre lichaam en de flexibele beweging niet kan bereikt worden.

Voor robots met flexibele gewrichten, is het model, onder bepaalde voorwaarden wel linearizeerbaar door niet–lineaire toestandsterugkoppeling. De belangrijkste voorwaarde is dat de kinetische energie van de rotor voornamelijk aan zijn eigen rotatie moet te wijten zijn. Aan deze voorwaarde is voldaan indien de overbrengingsverhouding tussen motor en gelid hoog is.

De niet–lineaire ontkoppelingsmethode werd toegepast voor de besturing van robots met flexibele gewrichten in 1987 door Bortoff en Spong [9], Forrest–Barlach en Babcock [26] en in 1988 door De Luca [15]. Deze studies geven enkel simulatieresultaten. Een van de redenen hiervoor is dat de toestandsvariabelen van het lineaire ontkoppelde deelsysteem (de positie, snelheid, versnelling en ruk van het gelid) niet meetbaar zijn. Om deze methode toe te passen is er dus nood aan een niet–lineaire toestandsschatter.

Hoofdstuk 2: modellering

Niet–lineaire besturing van flexibele robots is gebaseerd op een analytisch model dat het dynamisch gedrag van de robot beschrijft. Daarom wordt in hoofdstuk 2 een dynamisch model van robots met verscheidene flexibele geledingen en/of gewrichten afgeleid, gebruik makend van de methode van Lagrange. Het model is gekenmerkt door sterk niet–lineaire vergelijkingen met niet–lineaire koppelingen tussen de variabelen.

De doorbuigingen van de flexibele geledingen worden uitgedrukt

als

$$v_i(x_i,t) = \sum_{r=1}^{n_i} \Phi_{i,r}(x_i) q_{l\,i,r}(t)$$

met n_i het aantal modes gebruikt om de doorbuiging van gelid i te beschrijven, $q_{l\,i,r}(t)$ de tijdsafhankelijke veralgemeende coördinaten geassocieerd met de flexibiliteit van het gelid, en $\Phi_{i,r}(x_i)$ de functies die de modevormen van het flexibel gelid beschrijven.

De elastische koppeling tussen een gewricht en een gelid wordt gemodeleerd als een lineaire torsieveer.

Het uiteindelijke model is georganiseerd in een standaardvorm gelijkaardig aan het model voor een stijve robot. Deze standaardvorm wordt gegeven door:

$$\begin{bmatrix} \mathbf{M}_{11}(\boldsymbol{\theta}_l) & \mathbf{M}_{12}(\boldsymbol{\theta}_l) \\ \mathbf{M}_{12}^T(\boldsymbol{\theta}_l) & \mathbf{M}_{22}(\boldsymbol{\theta}_l) \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}}_0 \\ \ddot{\mathbf{q}}_f \end{bmatrix} + \begin{bmatrix} \mathbf{F}_{\mathbf{q}_0}(\mathbf{x}) \\ \mathbf{F}_{\mathbf{q}_f}(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} \mathbf{T} \\ 0 \end{bmatrix} \qquad (0.1)$$

met
$\mathbf{q}_0 = [q_{01} \ \ldots \ q_{0N}]^T \in \Re^N$ de vector van motorposities (stijve coördinaten)
en
$\mathbf{q}_f = [q_{f1} \ \ldots \ q_{fP}]^T \in \Re^P$ de vector van veralgemeende coördinaten geassocieerd met de flexibiliteit van de gewrichten en/of geledingen.

De massamatrix

$$\mathbf{M}(\boldsymbol{\theta}_l) = \begin{bmatrix} \mathbf{M}_{11}(\boldsymbol{\theta}_l) & \mathbf{M}_{12}(\boldsymbol{\theta}_l) \\ \mathbf{M}_{12}^T(\boldsymbol{\theta}_l) & \mathbf{M}_{22}(\boldsymbol{\theta}_l) \end{bmatrix}$$

wordt verondersteld enkel afhankelijk te zijn van de vector bevattende de posities van de geledingen $\boldsymbol{\theta}_l = [\theta_{l1} \ \ldots \ \theta_{lN}]^T \in \Re^N$.

De massamatrix is een symmetrisch positief definiete matrix, zijn inverse is ook symmetrisch en positief definiet en is gedefinieerd als

$$\mathbf{W}(\boldsymbol{\theta}_l) = \mathbf{M}^{-1}(\boldsymbol{\theta}_l) = \begin{bmatrix} \mathbf{W}_{11}(\boldsymbol{\theta}_l) & \mathbf{W}_{12}(\boldsymbol{\theta}_l) \\ \mathbf{W}_{12}^T(\boldsymbol{\theta}_l) & \mathbf{W}_{22}(\boldsymbol{\theta}_l) \end{bmatrix}$$

De toestandsvector wordt genoteerd als:

$$\mathbf{x} = [\mathbf{q}_0 \ \ \mathbf{q}_f \ \dot{\mathbf{q}}_0 \ \ \dot{\mathbf{q}}_f]^T$$

Hoofdstuk 3: Lineaire identificatie en besturing van een flexibele éénarmige robot

Identificatie en besturing van een flexibele éénarmige robot is een eerste stap in het ontwerp van een besturing voor een complexere konfiguratie. Het model dat de dynamica van een flexibele éénarmige robot, bewegend in een horizontaal vlak, beschrijft is lineair. Klassieke lineaire identificatie- en besturingstechnieken kunnen gebruikt worden.

Hoofdstuk 3 geeft een overzicht van lineaire identificatie en besturing. Een fysisch model van een flexibele éénarmige robot wordt afgeleid. In tegenstelling tot voor niet–lineaire besturing van robots met meerdere geledingen, is het voor lineaire besturing niet nodig de parameters uit het fysisch model te identificeren. Een mathematisch ingangs-/uitgangsmodel, zonder referentie naar de fysische achtergrond, volstaat. Er wordt besproken hoe een lineair model geidentificeerd wordt uitgaande van de kleinste kwadraten schatting van frekwentie responsie functies.

De beschreven besturing bestaat uit een voorwaartse koppeling van de versnelling en een terugkoppeling van het verschil tussen de gewenste en de gemeten systeemtoestand. De voorwaartse koppeling, tesamen met een geschikte berekening van de gewenste toestanden, heeft tot doel de volgfouten te minimaliseren.

In de meeste testopstellingen is de niet–lineaire Coulombwrijving niet verwaarloosbaar. Er wordt beschreven wat de invloed is van niet–lineaire wrijving op een lineaire besturing. Het veroorzaakt positiefouten, die niet kunnen gecorrigeerd worden enkel door toestandsterugkoppeling. Niet–lineaire wrijving veroorzaakt eveneens een servo–instabiliteit die aangeduid wordt met de term "limietcycli".

Om de positiefouten veroorzaakt door niet–lineaire wrijving te elimineren wordt integrale terugkoppeling gebruikt. Limietcycli worden vermeden door de bandbreedte van het gesloten kring systeem te beperken. In hoofdstuk 3 wordt eveneens een strategie voorgesteld om de limietcycli te beperken door de terugkoppeling geleidelijk uit te schakelen wanneer de gewenste positie bereikt wordt.

Hoofdstuk 4: De veralgemeende niet–lineaire ontkoppelingstheorie

Het gebruik van standaard niet–lineaire besturingsmethodes is beperkt zodra de bandbreedte van de besturing de helft van de laagste resonantiefrekwentie bereikt.

Toepassing van niet–lineaire ontkoppeling op de volledige systeemdynamica van flexibele robots (dynamica van het starre lichaam + flexibele dynamica) brengt geen oplossing omdat er onvoldoende onafhankelijke besturingsingangen zijn om alle niet–lineariteiten in het systeem te compenseren (er is slechts één actuator per gewricht, maar er zijn een oneindig aantal vrijheidsgraden). De onafhankelijke besturingsingangen kunnen gebruikt worden om de dynamica van het starre lichaam te ontkoppelen en te linearizeren, maar de flexibele dynamica wordt dan onbestuurbaar.

Voor robots met flexibele gewrichten kan er, onder bepaalde voorwaarden, een oplossing gevonden worden door gebruik te maken van de standaard niet–lineaire ontkoppelingstheorie. De bekomen besturingsmethode heeft echter het nadeel dat, buiten de posities en snelheden van de geledingen, ook de versnellingen en rukken van de geledingen nodig zijn voor terugkoppeling.

Het in deze thesis ontwikkelde en experimenteel geverifieerde besturingsalgoritme is gebaseerd op het idee van Froment en Faillot [29] en is gelijkaardig aan het algoritme ontwikkeld door Singh en Schy [64] en Pfeiffer en Gebler [55].

Het besturingsalgoritme is een veralgemening van de niet–lineaire ontkoppelingsmethode, zodanig dat deze methode ook toepasbaar is op robots met flexibele geledingen, en wordt daarom de veralgemeende niet–lineaire ontkoppelingsmethode genoemd.

De veralgemeende niet–lineaire ontkoppelingsmethode bestaat uit drie delen:
In een eerste deel wordt door de besturing een ontkoppelingskoppel geleverd dat niet–lineaire termen in de dynamica van het starre lichaam compenseert. Dit compensatiekoppel wordt berekend uitgaande van het dynamisch model van de flexibele robot en levert het grootste gedeelte van het koppel nodig voor het volgen van een trajekt.
In een tweede deel wordt een lineaire terugkoppeling van alle toestandsvariabelen (stijve en flexibele coördinaten en hun afgeleiden) toegepast. Hiervoor moeten de overblijvende niet–lineariteiten geli-

nearizeerd worden. De lineaire terugkoppeling verzekert de gepaste stabiliteit van het systeem rond de gewenste toestand. Ze wordt verondersteld kleine afwijkingen in het volgen van een trajekt te corrigeren en opgewekte trillingen te stabilizeren.
In een derde deel wordt statische en dynamische doorbuiging gecompenseerd door de flexibiliteit van de structuur in rekening te brengen bij de berekening van de gewenste toestanden. Hierdoor wordt het volgen van een referentietrajekt verbeterd. Deze compensatie wordt bekomen door de gewenste waarden van alle toestandsvariabelen te berekenen uitgaande van het model van de flexibele robot.

De veralgemeende niet–lineaire ontkoppelingsmethode kan zowel toegepast worden op robots met flexibele geledingen als op robots met flexibele gewrichten. Appendix B vergelijkt de veralgemeende niet–lineaire ontkoppelingsmethode, toegepast op robots met flexibele gewrichten, met de niet–lineaire ontkoppelings- of "computed–torque"–oplossing, zoals voorgesteld door Bortoff en Spong [9], en de singuliere perturbatie methode, voorgesteld door Spong [65]. De methodes verschillen in de keuze van de variabelen die teruggekoppeld worden. De veralgemeende niet–lineaire ontkoppelingsmethode heeft het voordeel dat metingen van de versnellingen en de rukken van de geledingen niet vereist zijn.

Wiskundige formulering van de veralgemeende niet–lineaire ontkoppelingsmethode

Gegeven is een gewenst traject dat moet gevolgd worden door de flexibele robot. De overeenkomstige gewenste toestandsvector wordt genoteerd als

$$\mathbf{x}_d = [\mathbf{q}_{0d} \;\; \mathbf{q}_{fd} \; \dot{\mathbf{q}}_{0d} \;\; \dot{\mathbf{q}}_{fd}]^T$$

De fout $\Delta\mathbf{x} = \mathbf{x} - \mathbf{x}_d$ is gedefinieerd als

$$\Delta\mathbf{x} = [\Delta\mathbf{q}_0 \;\; \Delta\mathbf{q}_f \; \Delta\dot{\mathbf{q}}_0 \;\; \Delta\dot{\mathbf{q}}_f]^T$$

De besturingswet is opgesplitst in twee terugkoppelkringen:
Eerste kring: de binnenste terugkoppelkring (niet lineair) realizeert linearizatie en ontkoppeling van de dynamica geassocieerd met de stijve coördinaten:

$$\mathbf{T} = \mathbf{F}_{\mathbf{q}_0}(\mathbf{x}) + \mathbf{W}_{11}^{-1}\mathbf{W}_{12}(\theta_l)\,\mathbf{F}_{qf}(\mathbf{x}) + \mathbf{W}_{11}^{-1}(\theta_l)\mathbf{T}_1 \qquad (0.2)$$

Vergelijking 0.1 wordt dan:

$$\begin{aligned} \ddot{\mathbf{q}}_0 = \quad & \mathbf{T}_1 && (lineair)\ (0.3) \\ \ddot{\mathbf{q}}_f = \quad & (\mathbf{W}_{12}^T\mathbf{W}_{11}^{-1}\mathbf{W}_{12} - \mathbf{W}_{22})(\theta_l)\,\mathbf{F}_{\mathbf{q}_f}(\mathbf{x}) + \mathbf{W}_{12}^T && \mathbf{W}_{11}^{-1}(\theta_l)\mathbf{T}_1 \\ & && (niet\ lineair) \end{aligned}$$

Tweede kring: de buitenste kring (lineair), verzekert de gewenste stabiliteit van het systeem rond de gewenste toestand $\mathbf{x}_d$, door lineaire terugkoppeling van de volledige toestand $\mathbf{x}$:

$$\mathbf{T}_1 = \ddot{\mathbf{q}}_{0d} - \mathbf{K}(\mathbf{x} - \mathbf{x}_d) \tag{0.4}$$

waar $\mathbf{K} = [\mathbf{K}_{\mathbf{q}_0}\ \mathbf{K}_{\mathbf{q}_f}\ \mathbf{K}_{\dot{\mathbf{q}}_0}\ \mathbf{K}_{\dot{\mathbf{q}}_f}]$ een N x (2N + 2P) matrix is die de terugkoppelconstanten bevat.
De gesloten kring vergelijkingen van het systeem worden bekomen door combinatie van vergelijking 0.4 en 0.3:

$$\begin{aligned} \Delta\ddot{\mathbf{q}}_0 + \quad & \mathbf{K}\Delta\mathbf{x} = 0 && (lineair) \\ \Delta\ddot{\mathbf{q}}_f + \quad & \mathbf{W}_{12}^T\mathbf{W}_{11}^{-1}(\theta_l)\,\mathbf{K}\,\Delta\mathbf{x} = && (0.5) \\ & (\mathbf{W}_{12}^T\mathbf{W}_{11}^{-1}\mathbf{W}_{12} - \mathbf{W}_{22})(\theta_l)\,\mathbf{F}_{\mathbf{q}_f}(\mathbf{x}) \\ & -\ddot{\mathbf{q}}_{fd} + \mathbf{W}_{12}^T\mathbf{W}_{11}^{-1}(\theta_l)\,\ddot{\mathbf{q}}_{0d} && (niet\ lineair) \end{aligned}$$

De volledige besturingswet wordt bekomen gebruik makend van vergelijkingen 0.2 en 0.4 als

$$\mathbf{T} = \mathbf{F}_{\mathbf{q}_0}(\mathbf{x}) + \mathbf{W}_{11}^{-1}\mathbf{W}_{12}(\theta_l)\,\mathbf{F}_{\mathbf{q}_f}(\mathbf{x}) + \mathbf{W}_{11}^{-1}(\theta_l)\ddot{\mathbf{q}}_{0d} - \mathbf{W}_{11}^{-1}(\theta_l)\mathbf{K}(\mathbf{x} - \mathbf{x}_d) \tag{0.6}$$

Figuur 0.1 toont het overeenkomstige blokdiagramma.

Berekening van het niet–lineaire compensatiekoppel: voorwaartse koppeling versus terugkoppeling

Het eerste deel van de regelaar bestaat uit een compensatiekoppel dat zorgt voor linearizatie en ontkoppeling van de dynamica geassocieerd met de stijve coordinaten.

Het compensatiekoppel kan berekend worden als een niet–lineaire voorwaartse koppeling of als een niet–lineaire terugkoppeling. Bij niet–lineaire voorwaartse koppeling worden de compensatiekoppels berekend op basis van de gewenste trajekten van de toestandsvariabelen

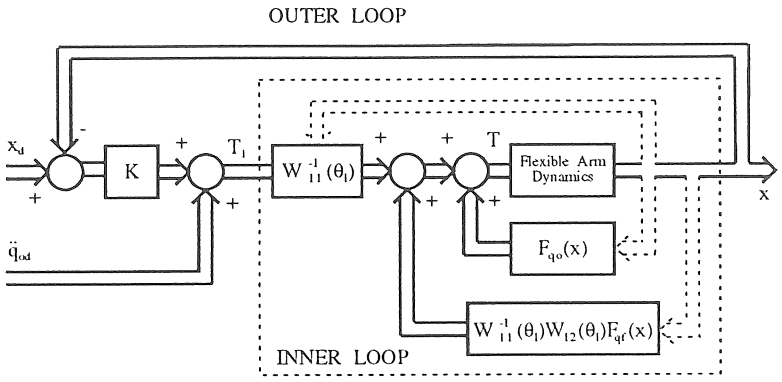

Figure 0.1: Blokdiagramma van de niet–lineaire besturing

terwijl bij niet–lineaire terugkoppeling de compensatiekoppels berekend worden uitgaande van de werkelijk optredende waarden.

De compensatie van de dynamica in de voorwaartse koppeling heeft het voordeel dat de compensatiekoppels op voorhand kunnen berekend worden. Tijdens de besturing worden ze dan opgeteld met de koppels berekend door de lineaire terugkoppeling. Een ander voordeel, dat geobserveerd werd tijdens de experimenten, is dat de besturing zich stabieler gedraagt bij het gebruik van ruis–vrije gewenste waarden. Om deze redenen wordt in de experimenten de voorwaartse koppeling gebruikt.

Berekening van de terugkoppelkonstanten

Een tweede deel van de veralgemeende niet–lineaire ontkoppelingsmethode is een lineaire terugkoppeling van alle toestandsvariabelen (stijve en flexibele coördinaten en hun afgeleiden). Om lineaire technieken (poolplaatsing, optimale controle) te gebruiken voor de berekening van de terugkoppelkonstanten moet het model gelinearizeerd worden rond het gewenste trajekt. De linearizatie kan gebeuren in een aantal punten langs het trajekt. De terugkoppelkonstanten zijn dan positieafhankelijk. Dit vraagt veel rekentijd. Een andere werkwijze is de linearizatie enkel uit te voeren in het eindpunt van het trajekt. Het belangrijkste doel van de lineaire terugkoppeling is de trillingen in het eindpunt van het trajekt zo snel mogelijk te dempen. Daarom is deze werkwijze verantwoord, indien het systeem niet onstabiel wordt

langs het trajekt. In praktische implementaties is het ook mogelijk een aantal terugkoppelkonstanten te tabelleren in funktie van de robotconfiguratie.

Wanneer de compensatie wordt toegepast in de voorwaartse koppeling is het in principe noodzakelijk het volledige model 0.1 te linearizeren (terugkoppeling kan de systeemdynamica veranderen, dit is niet het geval voor voorwaartse koppeling). Wanneer niet–lineaire compensatie wordt toegepast in de terugkoppeling, is het model reeds gedeeltelijk gelinearizeerd na het toepassen van de eerste kontrolekring. Voor de berekening van de terugkoppelkonstanten moet slechts dit gedeeltelijk gelinearizeerd model gelinearizeerd worden.

Berekening van de gewenste trajecten

Een derde deel van de veralgemeende niet–lineaire ontkoppelingsmethode compenseert de statische en dynamische doorbuiging. Hierdoor wordt het volgen van een referentietrajekt verbeterd.

Compensatie van statische en dynamische doorbuiging wordt bekomen door de gewenste waarden van alle toestandsvariabelen te berekenen uitgaande van het model van de flexibele robot. Hiervoor wordt, in een eerste stap, het trajekt van het roboteindpunt omgevormd naar angulaire posities, snelheden en versnellingen van virtuele stijve geledingen. De gewenste waarden van alle toestandsvariabelen worden dan berekend zodanig dat dezelfde positie van het roboteindpunt bekomen wordt als bij de robot met virtuele stijve geledingen.

Hoofdstuk 5: practische beschouwingen

Een belangrijke bijdrage van deze verhandeling is de praktische toepassing van het voorgestelde besturingsalgoritme, eerst op testopstellingen, daarna op een industriële robot.

Alvorens de veralgemeende niet–lineaire ontkoppelingsmethode toe te kunnen passen in praktijk, moeten de volgende praktische problemen behandeld worden;

- identifikatie van de modelparameters,
- keuze van de sensoren,
- schatting van de niet–meetbare toestandsvariabelen,

- berekening van de gewenste waarden van alle toestandsvariabelen,
- keuze van de orde van de polynomen gebruikt in de trajektgeneratie,
- compensatie van de niet–lineaire Coulombwrijving, zodanig dat de nadelige invloed van deze Coulombwrijving op de prestaties van de regelaar zo veel mogelijk geëlimineerd wordt.

Identifikatie wordt in deze verhandeling niet behandeld in een apart hoofdstuk. De gebruikte identifikatietechnieken worden uitgelegd in de hoofdstukken die de experimentele resultaten bespreken (hoofdstuk 6, 8 en 9). Dit heeft het voordeel dat de technieken rechtstreeks geïllustreerd worden door de voorbeelden. De andere problemen worden besproken in hoofdstuk 5. Er wordt eveneens beschreven hoe de verschillende delen van de besturing georganizeerd worden in de implementatie van de volledige besturing.

Een bijkomend probleem, geassocieerd met de implementatie op experimentele robots, zijn de rekenmogelijkheden van de beschikbare microprocessoren. Met de in de experimenten gebruikte processor kan de hoge bemonsteringssnelheid, nodig voor besturing in continue tijd, niet bereikt worden. Daarom moet de besturing uitgevoerd worden in discrete tijd. Bij de praktische implementatie wordt de rekentijd zoveel mogelijk beperkt door het zoeken naar eenvoudige oplossingen. Op dit moment zijn echter reeds snellere processoren beschikbaar. Dit biedt de mogelijkheid hogere bemonsteringssnelheden te kiezen.

Identificatie

De veralgemeende niet–lineaire ontkoppelingsmethode gaat uit van een dynamisch model van de flexibele robot. De ongekende parameters in dit model moeten geïdentificeerd worden. Deze modelparameters zijn bijvoorbeeld de inertiematrix, massa's, dempingskonstanten, veerkonstanten, eigenfrequenties, dempingsverhoudingen, enzovoort. Hoe nauwkeuriger het model, hoe beter de prestaties van de besturing (vooral met betrekking tot de volgnauwkeurigheid). Daarom wordt er in deze verhandeling veel aandacht besteed aan de identificatie van de modelparameters.

Voor een flexibele éénarmige robot kunnen de klassieke lineaire

"black box" identificatietechnieken, besproken in hoofdstuk 3, gebruikt worden. Voor robots met meerdere geledingen is een expliciete identificatie van de ongekende parameters van het niet–lineaire dynamisch model noodzakelijk.

Voor zelfgebouwde testopstellingen kunnen de verschillende delen van de robot apart genomen en afzonderlijk geïdentificeerd worden. Ook kunnen parameters uit het model berekend worden uitgaande van nauwkeurig gekende karakteristieken en geometrische eigenschappen van de opstelling. Voor industriële robots is dat gewoonlijk niet mogelijk, en zijn experimentele identificatietechnieken noodzakelijk.

In deze verhandeling worden verschillende identificatiemethodes gebruikt;
Een eerste methode gebruikt de "klassieke" identificatieschema's gebaseerd op een lineair ingangs-/uitgangsmodel. Hiervoor wordt de robot in een bepaalde stand gezet en geëxciteerd in een beperkt positiegebied. Parameterschattingen kunnen bekomen worden door het geïdentificeerde lineaire ingangs-/uitgangsmodel te vergelijken met het dynamisch model van de robot, gelinearizeerd rond die bepaalde stand. Om de positieafhakelijkheid te identificeren moet de procedure herhaald worden in verschillende standen.
Een tweede methode start rechtstreeks van de niet–lineaire bewegingsvergelijkingen. Het model wordt geschreven als

$$\Psi_1(\mathbf{x}, \dot{\mathbf{x}}, \ddot{\mathbf{x}})\, \theta_0 = \Psi_2(\mathbf{u}, \mathbf{x}, \dot{\mathbf{x}}, \ddot{\mathbf{x}})$$

met θ_0 de ongekende parametervector, en $\Psi_1(\mathbf{x}, \dot{\mathbf{x}}, \ddot{\mathbf{x}})$, $\Psi_2(\mathbf{u}, \mathbf{x}, \dot{\mathbf{x}}, \ddot{\mathbf{x}})$, een gekende matrix en vector respectievelijk, met elementen afhangend van de ingangs– en uitgangsmetingen, $\mathbf{u}$ en $\mathbf{x}$, en hun afgeleiden. De onbekende parametervektor, θ_0, kan dan geschat worden gebruik makend van standaard lineaire schattingstechnieken.

Bij de identificatie van de verschillende testopstellingen blijkt dat, om betrouwbare parameterschattingen te bekomen, het nodig is het identificatieprobleem op te delen in kleinere deelproblemen. Een mogelijkheid is de robot te demonteren in zijn verschillende onderdelen. Andere mogelijkheden zijn, eerst de parameters te identificeren die gerelateerd zijn met het dynamisch gedrag van een stijve robot, slechts één as van de robot te exciteren terwijl de positie van de andere assen constant gehouden wordt, enzovoort.

Het modelleren van de niet–lineaire Coulombwrijving verbetert de nauwkeurigheid van de parameterschattingen.

Bij de identificatie van de verschillende testopstellingen wordt telkens gebruik gemaakt van verschillende identificatietechnieken. Dit geeft een idee van de nauwkeurigheid van de bekomen parameters. Ook worden de experimenteel bekomen parameters vergeleken met analytische parameters, berekend uitgaande van gekende geometrische eigenschappen van de opstelling, en met parameters gegeven in ingenieurshandleidingen.

Keuze van de sensoren

Om de motorpositie te meten worden, in de experimenten, encoders gebruikt.

Voor robots met flexibele gewrichten zijn er extra sensoren nodig om de doorbuiging in de gewrichten te meten. Voor de testopstelling met twee geledingen en een flexibel eerste gewricht wordt de flexibiliteit van het gewricht gemeten met een LVDT (lineair variabele differentiaal transformator). Deze meting geeft rechtstreeks het verschil tussen de motorpositie en de positie van het gelid. Voor de KUKA IR 161/60 industriële robot zijn er extra encoders gemonteerd op de geledingen. De doorbuiging wordt dan berekend als het verschil tussen de gemeten motorpositie en de positie van het gelid.

Voor de éénarmige flexibele robot worden rekstrookjes gebruikt om de doorbuiging van het gelid te meten. In de experimenten wordt enkel de eerste flexibele mode beschouwd. De flexibele coördinaat is evenredig met het gemeten rekstrookjessignaal.

In de gebruikte testopstellingen, zijn dus de enige toestanden die niet rechtstreeks gemeten kunnen worden de motorsnelheden en de afgeleiden van de flexibele coördinaten.

Toestandsschatting

In de besturingstheorie wordt er verondersteld dat alle toestanden gekend zijn. In praktijk is het niet mogelijk alle toestanden rechtstreeks te meten. De bedoeling van toestandsschatting is een benadering van de volledige toestandsvector te construeren uitgaande van de beschikbare metingen.

Voor lineaire systemen (de flexibele éénarmige robot) kan een klassieke lineaire toestandsschatter gebruikt worden.

Voor niet–lineaire systemen wordt in dit hoofdstuk een niet–lineaire toestandsschatter voorgesteld. De structuur van de toestandsschatter is gelijkaardig aan die van een lineaire lopende toestandsschatter. Het schatten van de volgende toestand gebeurt door het integreren van het volledige niet–lineaire model. Linearizatie van het model laat toe de terugkoppelmatrix te bepalen.

In de experimenten zijn de enige toestanden die niet rechtstreeks kunnen gemeten worden de motorsnelheden en de afgeleiden van de flexibele coördinaten. Een andere oplossing, dan het gebruik van een toestandsschatter, is deze snelheden te berekenen door toepassing van rechtstreekse numerische differentiatie.
In principe moet rechtstreekse differentiatie vermeden worden. De reden is dat numerische differentiatie een minder stabiel algoritme is dat zeer gevoelig is voor ruis van hoge frekwenties. Om de zware rekenbelasting van de niet–lineaire toestandsschatter te vermijden, kan het gebruik van numerische differentiatie toch de geschikte keuze zijn. Een oplossing voor het probleem van de gevoeligheid voor hoge frekwenties, is de hoge frekwenties in de gemeten signalen weg te filteren door laagdoorlaatfilters.

Een filter veroorzaakt een fazeverschuiving. Daarom is het nodig exact gekende digitale filters te gebruiken en deze filters aan het model toe te voegen.

Het gebruik van laagdoorlaatfilters is ook een oplossing voor het "spill–over" probleem. Onder "spill–over" verstaat men de nadelige invloed van niet gemodelleerde eigenmodes op de besturing. Een continu gelid heeft een oneindig aantal eigenfrekwenties en bijhorende modes. Slecht een eindig aantal modes kunnen gemodelleerd worden (in de experimenten wordt enkel de eerste flexibele mode gemodelleerd). Door het gebruik van laagdoorlaatfilters worden de niet–gemodelleerde hogere frekwenties in de gemeten signalen weggefilterd.

Berekening van de gewenste toestanden

In de besturing wordt de volgfout teruggekoppeld. De volgfout is het verschil tussen de werkelijke en de gewenste toestandsvektor

$$\Delta \mathbf{x} = \mathbf{x} - \mathbf{x}_d$$

De gewenste waarden van alle toestandsvariabelen moeten berekend worden, gegeven een gewenst trajekt van het roboteindpunt.

Deze toestandsvariabelen zijn, de motorpositie, de veralgemeende coördinaten geassocieerd met de flexibiliteit van de geledingen en/of gewrichten en hun afgeleiden.

In een eerste stap wordt het trajekt van het roboteindpunt omgevormd naar angulaire posities, snelheden en versnellingen van virtuele stijve geledingen. Dit is een invers kinematisch probleem waarvoor een methode is beschreven in appendix C.

De gewenste waarden van de toestandsvariabelen worden dan berekend, uitgaande van het dynamisch model van de flexibele robot, zodanig dat dezelfde positie van het roboteindpunt bekomen wordt als bij de robot met virtuele stijve geledingen. Daarvoor wordt de relatie tussen de positie van de virtuele stijve geledingen en de toestandsvariabelen van het flexibel model afgeleid.

Voor een robot met flexibele geledingen zijn de berekeningen complexer dan voor een robot met flexibele gewrichten. Voor een robot met flexibele gewrichten zijn de gewenste trajecten exact, in de veronderstelling dat het model exact is. Voor een robot met flexibele geledingen, moeten er een aantal benaderingen gemaakt worden. De reden hiervoor is, dat een exacte berekening zeer complex is en resulteert in oscillerende gewenste toestanden (opwekking van resonantiefrekwenties). Het gebruik van deze trajecten in de besturing zou oscillerende koppels en dus ongewenste trillingen veroorzaken.

De gemaakte benaderingen bestaan erin enkel statische (gravitatie) en centrifugale krachten en inertie- en corioliskrachten in rekening te brengen. De koppeling tussen flexibele coördinaten wordt verwaarloosd. In praktijk wordt een goede benadering bekomen omdat het grootste deel van de doorbuigingen door deze krachten veroorzaakt wordt.

De bekomen procedure wordt in hoofdstuk 7 gebruikt in de simulatie van de besturing van een robot met twee flexibele geledingen.

Keuze van de orde van de polynomen gebruikt in de trajektgeneratie

Om excitatie van de structuurdynamica te vermijden wordt een zacht verlopend trajekt van het roboteindpunt aangelegd. Dit trajekt wordt berekend gebaseerd op hogere orde polynomen (zie appendix C).

De keuze van de orde van de polynomen is een compromis tussen de zachtheid van het trajekt (en dus betere prestaties van de besturing) en de uitvoeringssnelheid. Door de orde van de polynomen te verhogen zal ook de maximale versnelling die de actuator moet bereiken tijdens de beweging verhogen. Verzadiging van motoren beperkt de maximaal bereikbare versnelling. Daarom moet de uitvoeringssnelheid verminderd worden.

De gewenste toestanden worden gebruikt in de berekening van het niet–lineaire compensatiekoppel. Discontinuïteiten in het compensatiekoppel moeten vermeden worden om ongewenste opwekking van resonantiefrekwenties te voorkomen. In paragraaf 5.4.3 wordt aangetoond dat om deze discontinuïteiten te vermijden, het trajekt gebaseerd moet zijn op een polynoom van ten minste negende orde.

Compensatie van de Coulombwrijving

In de meeste testopstellingen is de niet–lineaire Coulombwrijving niet verwaarloosbaar. Hoofdstuk 3 beschrijft de technieken om de positiefouten en "limietcycli", veroorzaakt door niet–lineaire Coulombwrijving, te beperken of te elimineren.

Indien de niet–lineaire wrijving kan geïdentificeerd worden kan echter een groot deel van zijn effect geëlimineerd worden door rechtstreekse compensatie. Deze compensatie wordt bekomen door een koppel naar de motoren te zenden dat de niet–lineaire Coulombwrijving opheft.

Deel II: Simulatie en experimentele resultaten

Een tweede deel van deze verhandeling bespreekt simulatie en experimentele resultaten van de veralgemeende niet–lineaire ontkoppelingsmethode op testopstellingen met stijgende complexiteit. Dit houdt in

- Experimentele resultaten op een flexibele éénarmige robot
- Simulatieresultaten op een robot met twee flexibele geledingen
- Experimentele resultaten op een robot met twee geledingen en een flexibel eerste gewricht

- Experimentele resultaten op een KUKA IR 161/60 industriële robot

Hoofdstuk 6: Experimentele resultaten op een flexibele éénarmige robot

In hoofdstuk 6 worden de experimenten op een flexibele éénarmige robot beschreven.

Het model, waarop de besturing gebaseerd is, is lineair. Het is het resultaat van een algoritme gebaseerd op de kleinste kwadraten benadering. Het model brengt één flexibele mode in rekening. Het totale model is verdeeld in twee deelmodellen in serie waarvan de parameters apart geschat worden. Het eerste model heeft als ingang het motorkoppel en als uitgang het rekstrookjessignaal (dit is evenredig met de flexibele coördinaat van de eerste flexibele mode). Het tweede model heeft als ingang het rekstrookjessignaal en als uitgang de motorpositie. De theoretische vorm van het model is gebaseerd op het fysische model van een flexibele éénarmige robot.

De besturing gebruikt geen direkte meting van de positie van het eindpunt maar schat deze positie uitgaande van de gemeten motorpositie en van de rekstrookjessignalen.

Voorwaartse koppeling van de acceleratie wordt ingevoerd om de volgfouten te reduceren tot verwaarloosbare waarden. De berekening van de voorwaartse koppeling is gebaseerd op het quasi–statisch gedrag van het systeem; enkel de dynamica van het stijve lichaam wordt beschouwd.

Integrale terugkoppeling wordt gebruikt om de positiefouten veroorzaakt door niet–lineaire wrijving te elimineren. Limietcycli worden beperkt door de terugkoppeling geleidelijk uit te schakelen wanneer de gewenste positie bereikt wordt.

De invloed van hogere frekwentie modes wordt beperkt door de uitgangssignalen te filteren met tweede orde digitale laagdoorlaatfilters. De filters verhogen de orde van het model.

De terugkoppelkonstanten worden berekend door poolplaatsing. De beste besturingsresultaten worden bereikt met een bandbreedte van de gesloten kring die iets hoger is dan de eerste resonantiefrekwentie van de structuur. Wanneer de bandbreedte van de besturing nog verhoogd wordt treden limietcycli op. Dit belet een nauwkeurige positionering.

De bekomen besturing laat een nauwkeurige positionering toe met beperkte doorschot en trillingen, eveneens zijn de volgfouten bij het volgen van een gewenst trajekt minimaal.

Hoofdstuk 7: Simulatieresultaten op een robot met twee flexibele geledingen

In hoofdstuk 7 wordt de veralgemeende niet–linearie ontkoppelingsmethode toegepast op een simulatiemodel van een robot met twee flexibele geledingen. Het model wordt afgeleid, gebaseerd op de vergelijkingen van Lagrange, zoals beschreven in hoofdstuk 2. Het simulatiemodel brengt één flexibele mode in rekening per gelid.

Eerst wordt het compensatiekoppel berekend als een niet–lineaire voorwaartse koppeling, daarna als een niet–lineare terugkoppeling. Beide methodes geven gelijkaardige resultaten.

In vergelijking met een klassieke niet–lineaire besturing voor stijve robots worden de opgewekte trillingen zeer snel gedempt. Door de flexibiliteit van de structuur in rekening te brengen in de trajektgeneratie, zoals beschreven wordt in hoofdstuk 5, wordt de volgnauwkeurigheid sterk verbeterd. Zowel statische doorbuiging, veroorzaakt door de zwaartekracht als dynamische doorbuiging veroorzaakt door centrifugale krachten en inertie– en corioliskrachten, worden gecompenseerd.

In deze simulaties wordt een perfect dynamisch model verondersteld en zowel het simulatiemodel als het model gebruikt voor het ontwerp van de besturing, beschouwen enkel de eerste flexibele mode. De gevoeligheid van het algoritme voor onnauwkeurigheden van het model en voor niet gemodeleerde modes met een hogere frekwentie, wordt dus niet behandelt. Eenvoudige simulaties met een licht gewijzigd model (een aantal parameters zijn gewijzigd met maximaal 50 %) tonen geen stabiliteitsprobleem, de volgfout verhoogt echter beduidend. De experimenten op de testopstellingen en op de industriele KUKA robot (hoofdstuk 6, 8 en 9) tonen aan dat modelleringsfouten, spill–over en ook een onnauwkeurige modellering van de niet–lineaire motorwrijving, de bereikbare bandbreedte van de besturing beperken.

Hoofdstuk 8: Experimentele resultaten op een robot met twee geledingen en een flexibel eerste gewricht

Hoofdstuk 8 bespreekt de identificatie en besturingsresultaten van een robot met twee geledingen en één flexibel gewricht. De testopstelling bestaat uit een seriecombinatie van twee balken aangedreven door "direct drive" motoren. De eerste balk is door middel van veren met de motor verbonden. Dit simuleert een flexibel gewricht.

Eerst wordt het niet–lineaire model dat het dynamisch gedrag van het systeem beschrijft geïdentificeerd. De relatief hoge niet–lineaire motorwrijving wordt in rekening gebracht in de identificatie. Dit resulteert in betrouwbaardere schattingen van de modelparameters. De niet–lineaire wrijving wordt ook gecompenseerd in de besturing. Om de snelheden te schatten wordt de niet–lineaire discrete toestandsschatter, beschreven in hoofdstuk 5, toegepast.

In de trajektgeneratie wordt de flexibiliteit van de structuur in rekening gebracht.

De experimentele resultaten tonen de superieure prestaties van deze besturing aan, in vergelijking met de klassieke besturing voor stijve robots. De volgfout wordt gereduceerd met een faktor 10 en de positiefout met een faktor 35.

Hoofdstuk 9: Experimentele resultaten op een KUKA IR 161/60 industriële robot

In de literatuur worden de meeste besturingsalgoritmes getest op geïdealiseerde en speciaal ontworpen testopstelingen. De reden is dat een nauwkeurig model van de robot vereist is. Industriële robots worden vermeden omdat zij verre van ideaal zijn en daarom moeilijk te modelleren. Ook is de flexibiliteit van industriële robots beperkt. Dit stelt de toepasbaarheid van deze besturingsalgoritmes in vraag.

In hoofdstuk 9 wordt de veralgemeende niet–lineaire ontkoppelingsmethode geïmplementeerd op een KUKA IR 161/60 industriële robot. De flexibiliteit van de KUKA robot is beperkt. De belangrijkste flexibiliteiten kunnen gelokaliseerd worden in de eerste drie gewrichten. De dynamica van de tweede en derde as is ontkoppeld van de dynamica van de eerste as. Daarom wordt de eerste as bestuurd, gebruik

makend van een lineaire positieafhankelijke toestandsbesturing [68], terwijl de tweede en derde as bestuurd worden gebruik makend van de veralgemeende niet–lineaire ontkoppelingsbesturing. De laatste drie gewrichten zijn stijf. Zij worden bestuurd gebruik makend van onafhankelijke PID-besturingen in snelheidssturing.

De flexibiliteit van de eerste drie gewrichten wordt gemodelleerd als een lineaire torsieveer. De identificatie van het dynamisch model wordt uitgevoerd in twee stappen: eerst worden de parameters geïdentificeerd die gerelateerd zijn met het dynamisch gedrag van een stijve robot, daarna worden de veerkonstanten geschat. De veerkonstanten worden geschat gebruik makend van vier verschillende meetmethodes; namelijk door middel van statische metingen, quasi–statische metingen, excitatie met een bandbeperkt randomsignaal en sinusoïdale excitatie. Een vergelijking van de resultaten geeft een idee van de bekomen nauwkeurigheid.

De experimentele resultaten van de veralgemeende niet–lineaire ontkoppelingsmethode worden vergeleken met de resultaten van een klassieke PID–besturing. In vergelijking met de klassieke PID-besturing is de maximale dynamische volgfout gehalveerd en is de statische fout veroorzaakt door de zwaartekracht gecompenseerd. Deze resultaten tonen aan dat, zelfs voor een industriële robot met beperkte flexibiliteit, de prestaties kunnen verbeterd worden.

Algemeen besluit

Flexibiliteit van gewrichten of geledingen beperkt de bereikbare prestaties van traditionele besturingen ontworpen voor stijve robots. Het doel van het onderzoek, waarover in deze verhandeling gerapporteerd wordt, is de traditionele besturingen aan te vullen, ten eerste om ongewenste trillingen te dempen of te voorkomen en ten tweede om het volgen van een gewenst trajekt te verbeteren. Hiervoor is een nieuw besturingsalgoritme ontwikkeld en experimenteel geverifiëerd op testopstellingen en op een industriële robot.

De auteur van deze verhandeling hoopt dat dit werk zal bijdragen tot de verbetering van de prestaties van traditionele robots waarvan de flexibiliteit van de overbrengingen een belangrijk probleem blijft. Eveneens, hoopt de auteur dat dit werk zal bijdragen tot de ontwikkeling van een nieuwe generatie van snelle en nauwkeurige robots. Het ge-

bruik van lichte armen, met verhoogde flexibiliteit, zal hierbij nodig zijn om hogere bewegingssnelheden, efficiënter gebruik van energie, veiligere operatie en verhoogde mobiliteit te bekomen.

Chapter 1

Introduction

1.1 Problem formulation

Industrial robots have to meet ever increasing specifications in terms of speed, acceleration and range along with improved accuracy. Structural flexibility is a primary limitation for manipulator performance.

Flexibility may be caused either by elasticities of one or more links, or by elasticities occuring at the joints, or by a combination of both. Traditionally, to obtain sufficient accuracy, industrial robots are built as stiff as possible. Even then flexibility of drive systems remains an important problem. The stiffness and weight of these traditional robots limit the achievable speed and acceleration. As a result, weight and inertia of moving parts have to be decreased, which leads to lower structural resonance frequencies.

Flexibility limits the achievable bandwidth of traditional controllers which are designed for rigid systems. The performance of these controllers deteriorates very rapidly when the closed loop bandwidth approaches the lowest structural resonance frequency. A "rule of thumb" in robot control is to limit the control bandwidth to less than one-half the lowest cantilever frequency (resonance of the tip with all the motors locked). The consequence is that structural oscillations are only weakly damped out by the controller. These osillations occur when exciting system resonances by moving the manipulator too brusquely.

This is the reason why flexibility limits the efficiency of point to point control. Before starting an operation at the destination point, it is necessay to wait until vibrations have died out. Furthermore, static

deflection or bending due to gravitational forces causes steady state errors between the actual tip position and the desired set point.

For tracking control applications, where the aim is to let the tip of the robot follow a desired trajectory as closely as possible, flexibility limits the tracking accuracy. The tracking errors are caused by the deflections due to inertial, coriolis and centrifugal forces. For example, to move from one point to another, with zero begin- and end velocity, the flexible manipulator is forced to accelerate and decelerate, which causes inertial forces, and hence results in corresponding deflections.

The focus of the research reported in this dissertation is to supplement traditional control, firstly to prevent or damp out undesired oscillations resulting from excitation of structural dynamics, and secondly to improve the tracking of a reference trajectory designed for a rigid robot. To this end new control algorithms are developed and verified experimentally, first on test set ups, then on an industrial robot.

1.2 Literature Review

After a short review of control of rigid manipulators, a state of the art in control of flexible manipulators is given.

When this research started, the state of the art in control of flexible manipulators was not very advanced. Only recently a growing interest exists in this topic. Therefore most of the cited studies are performed in parallel with the work reported in this dissertation.

Many authors consider only elasticity in the joints, which indeed represents the main cause of flexibility in present day industrial robots. Other authors study the effect of link elasticity, but in most cases both the derived dynamic model and the control approach are limited to a one–link robot. Extension to the multi–link case proves to be very difficult, because the dynamic model becomes very complex. The actual number of realistic approaches which are applicable to multi–link systems is therefore extremely limited. Very few experimental results have been published to date. Most of the studies present only analytical or simulation results.

A last section in this literature review presents an overview of research related to control of flexible manipulators. The controller, proposed in this dissertation, is based on an analytical model describing the flexible manipulator dynamics. Essential for model based control

system design is the knowledge of this model. To this end, modeling of flexible manipulators and identification of these models is reviewed. When implementing control algorithms on experimental robots, the problem of nonlinear motor friction has to be dealt with. To this end, an overview in techniques to minimize the influence of nonlinear motor friction on the controller, is given.

1.2.1 Control of rigid manipulators

Manipulators are multi–input, multi–output systems, described by highly nonlinear equations with nonlinear couplings between the variables of motion. However, the present commercial control systems make use of decoupled control [18], [48]. Strong simplifications are made in the system description; cross inertias and coriolis and centrifugal forces are neglected so that the multi–input, multi–output system appears as a decoupled set of single–input, single–output systems. In addition the inertia and gravitation forces are assumed to be position independent, so that classical linear servo control can be applied. Because of these strong simplifications, the dynamic behaviour of the decoupled control systems is not optimal.

The use of decoupled control systems in industry has several reasons [48]. The knowledge of a dynamic robot model, which has to be identified by means of experimental identification procedures, is not required. The implementation of these decoupled control systems relies on rather straightforward rules. Hence, it can be performed easily. In addition the hardware requirements, concerning processing speed, sampling rate and memory volume are low.

The advanced manufacturing problems of the next years will demand increased performance, especially concerning velocity and dynamic accuracy, which can no longer be met by the present commercial control systems. In the last 15 years a large number of control algorithms for rigid robots has been developed in the robot control research area [19], [48].

The most advanced controllers are based on nonlinear decoupling (computed torque methods). By compensation of the nonlinear terms in the rigid body dynamics, the system is reduced to a set of linear decoupled subsystems (1 subsystem per link). Each of these subsystems is then controlled by a classical decoupled linear feedback loop given

by

$$(\ddot{\theta}_{ld} - \ddot{\theta}_l) + \mathbf{K}_v(\dot{\theta}_{ld} - \dot{\theta}_l) + \mathbf{K}_p(\theta_{ld} - \theta_l) = 0$$

where $\mathbf{K_p}$ and $\mathbf{K}_v$ are diagonal matrices representing position and velocity feedback gains, θ_l is the vector of link positions and θ_{ld} is the vector of desired link positions. The first of these methods has been developed originally in the mid–1970's by Freund [28].

Two basically different decoupling schemes are possible [44]. The principle of feedback decoupling determines the actuating force, according to the nonlinear model, depending on the measured positions and velocities. The principle of feedforward decoupling determines the actuating force, according to the nonlinear model, as a function of the desired trajectory and their derivatives.

The achievable performance of these nonlinear decoupling controllers is limited by the modeling accuracy. A way to avoid the problems of model uncertainty is to apply adaptive control. The proposed concepts can be subdivided into two groups; model reference adaptive control [39], [22] and the self–tuning adaptive control [39], [45]. However, the high on–line computation effort and the uncertain question of stability questions their practical implementation [39].

1.2.2 Control of flexible manipulators

Control of manipulators with one link has to be distinguished from control of manipulators with two or more links. Since the model describing the dynamic behaviour of a one–link flexible robot is linear, classical linear identification and control techniques can be used.

The first experiments on noncollocated endpoint control of a very flexible one–link manipulator are published by Cannon and Schmitz in 1984 [11]. Afterwards several researchers have treated this problem [41], [58], [60], [61], [70], [73], [80], [83].

The next logical step in the progression of experiments is control of flexible manipulators with more useful multi–link geometries. Models of multi–link flexible robots are characterised by highly nonlinear equations with nonlinear couplings between the variables of motion. The more complex problem of control of multi–input multi–output systems with highly nonlinear dynamics has to be investigated.

The presented methods can be classified into three groupes: the computed torque methods, the singular perturbation methods and the multistage approaches. Experimental results are reviewed separately,

because of their limited appearance in literature.

Computed torque methods (nonlinear decoupling, nonlinear state feedback, inverse dynamics)

A decoupling and linearizing algorithm is used to compute the torque to be applied at each motor in order to make the manipulator appear as a linear multivariable system for which a simple linear controller is then designed. In other words, given the nonlinear system,

$$\dot{\mathbf{x}} = \mathbf{A}(\mathbf{x}) + \mathbf{B}(\mathbf{x})\mathbf{u},$$

where $\mathbf{x}$ is the n–dimensional state vector and $\mathbf{u}$ is the m-dimensional input vector, the decoupling and linearizing consist of finding a coordinate transformation

$$\mathbf{y} = \mathbf{F}(\mathbf{x}),$$

and a nonlinear feedback

$$\mathbf{u} = \mathbf{a}(\mathbf{x}) + \mathbf{B}(\mathbf{x})\mathbf{v},$$

such that the transformed state $\mathbf{y}$ satisfies the linear system

$$\dot{\mathbf{y}} = \mathbf{A}_l\mathbf{y} + \mathbf{B}_l\mathbf{v}$$

This general nonlinear decoupling theory is developed by Freund [28]. For the class of rigid systems it is known that, under the assumption of perfect rigidity, the system is globally linearizable via nonlinear state feedback. The nonlinear decoupling controller for rigid manipulators, described in the previous section 1.2.1, is obtained.

In contrast to rigid robots, it can be proved that, when taking link flexibility into account, global linearization of combined rigid body and flexible motion cannot be achieved [74].

On the other hand, for flexible joint robots, Bortoff and Spong [9] pointed out some basic assumptions which lead to a model which is globally linearizable by nonlinear state feedback. The main assumption is that the kinetic energy of the ith motor due to the motion of all previous links in the robot chain is neglected. This means that cross–coupling terms due to motor mass are not taken into account in the robot inertia matrix. It can be shown that this assumption is only meaningful for high gear transmission rates.

Computed torque solutions for some two and three link planar flexible joint manipulators have been reported in 1987 by Bortoff and Spong [9] and Forrest-Barlach and Babcock [26], and in 1988 by De Luca [15], [16]. The additional linear stabilizing feedback in the methods of Bortoff and Spong and of Forrest-Barlach and Babcock is given by

$$\theta_{ld}^{IV} + \mathbf{K}_3(\theta_{ld}^{III} - \theta_l^{III}) + \mathbf{K}_2(\ddot{\theta}_{ld} - \ddot{\theta}_l) + \mathbf{K}_1(\dot{\theta}_{ld} - \dot{\theta}_l) + \mathbf{K}_0(\theta_{ld} - \theta_l)$$

where the $\mathbf{K}_j$ (j=0,...,3) are constant diagonal matrices representing state error feedback gains, θ_l is the vector of link positions and θ_{ld} is the vector of desired link positions. It should be noted that, in contrast to the rigid case, the transformed state variables, θ_l, $\dot{\theta}_l$, $\ddot{\theta}_l$ and θ_l^{III}, are not measurable any more. Hence, in order to apply this method, there is a need for either a nonlinear state estimator, or measurements of accelerations of the link position and their derivatives (jerks), supposing link positions and velocities can be measured. The accelerations and jerks can also be computed by expressing them as functions of the other system states. This requires differentiation of the model. But 1) this yields an open loop estimate and 2) it implies an expensive on–line computation. Another possibility is to compute the accelerations and jerks by differentiation of the measured positions (or velocities). But this solution is not suited in practice, because of the sensitivity of the differentiation algorithms to high frequency noise.

Singular perturbation methods

The singular perturbation method is a mathematical method for separating nonlinear systems into coupled "slow" and "fast" subsystems. ([52], [65], [62], [42]). The resulting slow subsystem allows the determination of a tracking control as for rigid manipulators, since the number of control variables equals that of controlled variables. For the fast subsystem an additive control is in charge of stabilizing the deflections along the joint angle trajectory. Simulations for a one link manipulator with link flexibility or joint flexibility have been performed in 1987 by Spong and Kokotovic [65] and in 1988 by Siciliano and Book [62] and by Khorrami and Özgüner [42].

Multistage approaches

In 1985 Singh and Schy [64] presented an approach consisting of two stages. The first stage is a nonlinear decoupling control law which is derived using an inversion algorithm, such that in the closed loop system, independent control of joint angles is accomplished. In the second stage, for the stabilization of the elastic oscillations, a linear feedback control law is obtained by applying linear quadratic optimization to the linearized model.

In 1988 Pfeiffer and Gebler [55] proposed an approach consisting of three stages. Firstly, feedforward decoupling is applied to control the rigid body dynamics along an optimized reference trajectory. Deviations from this path due to structural elasticity are counterbalanced in a second step by a quasi–static feedforward control, and in a third step by an elasticity control using strain gauge measurements at the elastic links.

Experimental results

Wheras control of flexible manipulators has been studied extensively in simulations, the real time results have been lacking in the robotics literature. The few experimental results published to date are on specially developped test setups. Industrial robots are avoided.

In 1989, Hollars presented experimental results in endpoint control of manipulators with elastic drives [36], [37] performed on a two–link test setup with two flexible joints. The controller consists of a nonlinear state estimator combined with a full state feedback LQ regulator (without feedforward), designed in an optimum linearization point.

In 1987 Gebler [30] [31] presented experimental results on a specially designed three–link manipulator with elastic links and drives. The desired joint angles calculated under the assumption of rigid joint and links are modified by taking into account "nominal deflections". Gravitational and inertial effects are taken into account. By this feedforward control strategy that compensates for the elastic deformations the oscillations of the endpoint of the manipulator can be reduced significantly. This is combined with a feedforward decoupling control to control rigid body dynamics (only feedback of rigid coordinates). In 1988 a control scheme was added which fed back strain gauge measure-

ments at the elastic arms to damp out remaining oscillations (Pfeiffer and Gebler [55])

1.2.3 Related research

This section presents a short overview of research which is related to the work reported in this dissertation. This includes modeling of flexible manipulators, identification of the models and minimization of the influence of nonlinear motor friction on the controller.

Modeling of flexible manipulators

In 1984 Cannon and Schmitz [11] developed a model of a one–flexible link robot using the Lagrangian approach.

Because the derivation of the dynamic model of multilink flexible manipulators becomes very complex and tedious, most papers give only a general formulation of the model, the dynamic equations are not written out explicitly. In 1987 Gebler derived in his dissertation [30] a dynamic model of a robot with two flexible links and joints, using the principle of d'Alembert-Jourdain. In 1988 Benati and Morro [8] described the dynamics of a chain of flexible links. Oakly and Cannon [54] treated the problem of selecting mode shapes for two–link flexible manipulators. Recently new results are published by Cetinkunt and Book [14] on symbolic modeling of robotic manipulators with compliant links and joints.

Identification

For linear systems a lot of identification techniques are found in literature. A good overview is given by Swevers [67]. These linear techniques can be used for the identification of the model of a one–flexible link robot.

In contrast to the identification of linear systems where a black box approach can be used, the identification of nonlinear systems requires an explicit nonlinear dynamic model, with unknown parameters which have to be identified.

In literature, it is mostly assumed that the parameters can be calculated accurately enough from known characteristics and geometric properties [32]. However for industrial robots with complex geome-

tries, this does not always give reliable results.

Amstrong, Khatib and Burdick [5] have determined experimentally the mass, the center of mass and the inertia matrix of the individual links of a PUMA 560 robot.

In practice it is not always possible or practical to disassemble robotic manipulators into components. Therefore experimental identification methods, based on measurements performed on the entire robot, are needed. Even for rigid robots very few experimental methods are published. In 1985 Khosla and Kanada [43] presented an algorithm to estimate the dynamic parameters of a robot from the measurements of its inputs (actuating torques) and outputs (joint positions, velocities and accelerations). The nonlinear model is linearized about the nominal values of the dynamic parameters supposing the kinematic parameters are known. Nominal values of dynamic parameters are, for example, obtained from engineering drawings. Identification of the dynamic parameters is then a problem of linear estimation. It is proposed to lock the first i-1 joints mechanically to reduce the complexity of the model.
In 1988 Held and Maron [35] described the identification of friction characteristics, inertial and coupling coefficients in robotic drives, based on measurements of current and speed. The moment of inertia and the coupling coefficients are obtained by a correlation method, and an energy balance reveals the friction characteristics.
Wyckaert [81] described the identification of friction characteristics and inertial coefficients based on sinusoidal excitation techniques.

Minimizing the influence of nonlinear motor friction on the controller

Several methods for compensating Coulomb friction are available. One method is to wrap a very high–gain rate–loop around the motor and to use feedback to make the Coulomb friction disturbance torque very small outside the innermost loop.
A second method is to use adaptive pulse width control for precise positioning. When the motion stops due to stiction and a positioning error remains, a single pulse input, the width of which is a function of the error, is applied to move the system and eliminate the error. When the motion stops again and an error still remains, a functional

relationship between the pulse width and error is updated by an adaptive algorithm, and a new pulse is applied (Yang and Tomizuka, [82]).
Another method is to use a model of the friction to feedback a Coulomb–friction–canceling torque signal in the spirit of the computed torque methods (Kubo, Anwar and Tomizuka, [47], Vossoughi and Donath, [79]).
The modeling method can be implemented in anticipation of adaptive learning of the friction model (Canudas, [13]).
Hollars [37] combines Coulomb–friction–cancellation with a small dither and a small deadband around zero velocity. However, when using dithering, motion does not stop completely, bearing wear is accelerated, and power is wasted.

1.3 Results

The research reported in this dissertation makes the following contributions to the field of automatic control and robotics.

Development of a control algorithm applicable to flexible joint robots as well as flexible link robots

A continuous time control algorithm applicable to flexible joint robots as well as flexible link robots is proposed. It consists of

- a nonlinear feedforward or feedback to cancel out nonlinear terms in the rigid body dynamics.
- linearization of the remaining nonlinearities, and linear feedback of all state variables (rigid and flexible coordinates). This ensures proper stability of the system about the desired state.
- compensation of static and dynamic deflection. This improves the tracking of a reference trajectory. The compensation is achieved by calculating the reference values of all state variables based on the model of the flexible robot.

The compensation torque can be a nonlinear feedforward torque or a nonlinear feedback torque. If an exact model of the manipulator is available, then both the feedback and feedforward compensation

scheme will give similar results. Using the feedforward dynamic compensation may have some practical advantages; therefore, in the experiments, nonlinear feedforward compensation is applied. The feedforward dynamic compensation technique is based on the premise that the largest part of the torque for trajectory tracking is calculated using the inverse dynamic model in the feedforward path. This control signal is then augmented by the feedback signal which corrects for small deviations in the trajectory tracking, and stabilizes excited vibrations.

The nonlinear control problem is formulated as a generalization of linear state feedback control methods with additional feedforward torque to improve tracking accuracy. While in linear control the feedforward signal is calculated based on a linear model, in nonlinear control the feedforward calculation is more complex and based on a nonlinear physical model. While in linear control linearization is not needed, in nonlinear control the feedback gains are calculated based on the model linearized in a certain position.

The complexity of the techniques proposed in literature and of the computations involved has limited to a certain extent the understanding of the generality of the design. Evaluating the proposed control algorithms in literature shows that all algorithms are a variant of the same idea.

The algorithm proposed in this thesis can be classified as a multistage approach, similar to the method proposed by Pfeiffer [55] and Singh [64].

The algorithm is a variant, or genealization of the computed torque algorithms (calculation of the feedforward torque). Therefore the method is called "generalized nonlinear decoupling". In comparison with the computed torque algorithms, the algorithm is not only applicable to rigid or some classes of elastic–drive manipulators, but also to manipulators with link flexibility.

The algorithm can also be seen as a variant of the singular perturbation method. The same idea of separating the nonlinear system into a slow subsystem (control as for rigid manipulators) and a fast subsystem (additive control to stabilize the deflections) is used. Whereas for singular perturbation methods the mathematical formulation and the involved computations are complex, the mathematical formulation is more straightforward here.

Appendix B describes and compares the computed torque, the singu-

lar perturbation and the generalized nonlinear decoupling solution for flexible joint robots. For the three methods, using the feedforward compensation scheme, the same feedforward torque is obtained. The methods differ in the choice of feedback variables and in the way the feedback gains are calculated. The generalized nonlinear decoupling solution has the advantage, in comparison with the computed torque solution and the singular perturbation solution that the needed feedback variables can be estimated easily from the available measurements.

Modeling of flexible manipulators

A dynamic model of robotic manipulators with compliant joints and links is derived based on the Lagrangian assumed modes formulation. The final form of the equations is organized in a form similar to rigid manipulator equations. The model is written out explicitely for a one– and two–flexible link robot and for a two–link robot with flexibility in the joints.

Identification of flexible manipulator models

Since the design of the control system is based on a model of the actual manipulator, the achievable performance is limited by the modeling accuracy. This is the reason why a lot of attention is paid to the identification of the models.

For a one–link flexible robot linear "black box" identification techniques can be used. For multi–link flexible robots, it is necessary to identify the unknown parameters of an explicit nonlinear dynamic model.

For test set ups which have been built in–house, it is possible to take apart the different parts of the robot and to identify them separately. In addition, model parameters can be calculated from accurately known characteristics and geometrical properties. For industrial robots, this is usually not possible. Therefore experimental identification techniques are necessary.

This dissertation does not give a general one step identification procedure. Instead, an overview of methods, which have been proven useful in the identification of the different set ups, will be given. Prac-

tice has shown that splitting up the identification problem into smaller subproblems is a better approach than trying to identify all the parameters in one step. Using various alternative identification techniques gives an idea of the accuracy of the obtained parameters. Also, comparison of the experimentally obtained parameters with analytically calculated parameters obtained from geometrical and physical properties, and with parameters given in engineering drawings is useful.

Two different approaches are used.
A first approach uses the classical identification schemes based on a linear input–output model. To this end the robot is put in a certain position and is excited in a limited position range. If the model is linearized with respect to that position, parameters are obtained by comparing measured transfer functions with analytical transfer functions. It is necessary to repeat the procedure in different positions to identify the position dependency.
The second approach starts directly from the nonlinear equations of motion. Writing the model as

$$\Psi_1(\mathbf{x}, \dot{\mathbf{x}}, \ddot{\mathbf{x}})\ \theta_0 = \Psi_2(\mathbf{u}, \mathbf{x}, \dot{\mathbf{x}}, \ddot{\mathbf{x}})$$

with θ_0 the unknown parameter vector, and $\Psi_1(\mathbf{x}, \dot{\mathbf{x}}, \ddot{\mathbf{x}})$, $\Psi_2(\mathbf{u}, \mathbf{x}, \dot{\mathbf{x}}, \ddot{\mathbf{x}})$, a known matrix and vector respectively, with elements depending on the input and output measurements, $\mathbf{u}$ and $\mathbf{x}$, and their derivatives, the unknown parameter vector can be estimated by any standard linear estimation technique. Dividing the identification in different steps by, for example, disassembling the robot into components, identifying first parameters related to rigid body dynamics, performing motion of a single axis while the positions of the others are fixed, etc..., is necessary to obtain accurate estimates. Inclusion of the nonlinear Coulomb friction in the model improves the accuracy of the estimation.

Trajectory calculation

Based on the desired trajectory of the tip position, an exact trajectory is calculated for all the state variables (rigid coordinates as well as flexible coordinates). This calculation is based on the dynamic model of the flexible robot. For a manipulator with joint flexibility exact state trajectories can be calculated. For a manipulator with link flexibility, the computation is much more complex and intricate. To avoid oscillating trajectories, some approximations are made. These

approximations correspond to considering only quasi–static relationships. This means that only rigid body dynamics and deflections due to gravitational, inertial, coriolis and centrifugal forces are considered. Simulations on a two flexible link robot show that these approximations are justified.
By using these trajectories of the state variables in the controller the tracking errors are reduced significantly. B. Gebler [30] states that this control strategy also reduces the oscillations of the endpoint of the manipulator significantly.
In order to limit the excitation of the structure, it is important to use smooth trajectories. In this work polynomial trajectories are used for this purpose. The choice of the order of the polynomials will be discussed in detail.

Experimental considerations

The problems associated with implementation of the derived control algorithms on an experimental robot have been solved.

This includes the choice of sensors and estimation of unmeasurable state variables. The use of estimators has generally been ignored in manipulator control research to date. Most analytic or simulation studies make the assumption that all the desired states are available from noise-free sensors. A nonlinear constant gain estimator is proposed in this work and implemented on the experimental manipulator. However, for the proposed generalized nonlinear control method, the transformed state variables are the motor positions and flexible coordinates and their derivatives, so that in the experiments, the only states which cannot be measured directly are the motor velocities and the velocities of deflection. It is shown that calculation of velocities by direct numerical differentiation also gives good results with less calculation time, provided the measurements are filtered with digital low pass filters.
Low pass filtering is also needed to reduce spillover problems. The filter dynamics are taken into account in the control.

Another problem associated with experimental implementation is caused by the influence of dominant nonlinear friction. Nonlinear friction has a large effect on the design of a controller. It causes limit cycles and steady state errors which can not be corrected with plain

state feedback. In linear control design steady state errors are eliminated with integral feedback and limit cycles are avoided by restriction of the closed loop bandwith. In this dissertation an additional strategy is proposed to reduce limit cycles. It consists of phasing out the feedback once the final position is reached. In addition to these techniques, which do not need identification of the nonlinear friction, a motor Coulomb friction compensation scheme is implemented on the experimental manipulators. This reduces tracking errors, steady state errors and settling time.

A last problem associated with implementation on an experimental robot are the computational capabilities of available microprocessors. The measurement and control unit, used in the experiments presented in this thesis, is a VME–system based on a Motorola 68030 25 MHz microprocessor with a MC 68882 25 MHz floating point coprocessor. The computational capabilities of this system are not sufficient for high sample–rate control of continuous time controllers. Therefore discrete control design is used. Computation time is minimized by striving after simple and straightforward solutions. However, at the moment newer processors with increased computational capabilities are available. (For example, the computational capabilities of the last DSP 320C40 are at least 10 times higher.) Using these processors gives the possibility to choose higher sample–frequencies.

Application to a simulation model of a flexible two–link robot

The generalized nonlinear decoupling method is applied to a simulation model of a flexible two–link robot. The simulations show that, in comparison with a rigid controller, the excited vibrations are damped out very rapidly. They show also the importance of the desired state calculation. If the desired joint angles and flexible coordinates are calculated based on the total dynamic model, compensation of static and dynamic deflection is obtained and the tracking errors are reduced to negligible values.

Identification and control of a flexible one–link robot

Identification and control of a flexible one–link robot is a first step in the development of a controller for a more complex configuration.

Linear strategies, applied to the single degree of freedom, must be extended to nonlinear techniques for the more complex case. Therefore a first experiment is performed on a flexible one–link robot.

The linear controller is formulated in the same way as the proposed more general nonlinear controller. Feedforward is introduced to reduce tracking errors to negligible values. The calculation of the feedforward is based on the quasi–static behaviour of the system; only rigid body dynamics and deflections due to inertial forces are considered. The state feedback gains are calculated using a pole placement algorithm. The best control results are obtained for a closed loop bandwidth which is a little higher than the first resonance frequency of the structure. Integral control eliminates positioning errors, limit cycles are reduced by phasing out the feedback once the final position is reached. Observation spill–over is avoided by filtering the output signals using a second order digital filter.

The controller allows accurate tracking and positioning with very limited overshoot, oscillation or tracking errors.

Identification and control of an experimental two–link manipulator with a flexible first joint

Because of the problems of constructing a more complex multi–link experimental manipulator with distributed link flexibility, an experimental two–link manipulator with a flexible first joint is constructed. Flexible joint manipulators are an important class of systems, since they are used to study the performance limitations imposed by drive-train compliance on current commercial robots. The generalized nonlinear decoupling method is applied to this test setup. The nonlinear Coulomb friction is compensated in the control. The velocities are estimated using a nonlinear discrete state estimator. And the trajectory calculation algorithm takes the flexibility of the structure into account. The experiments show the superior performance of this controller compared to classical rigid control: the excited vibrations are damped out very rapidly, the tracking error is reduced by a factor 10, and the steady state error by a factor 35.

Identification and control of a KUKA IR 161/60 industrial robot

Experimental results on a KUKA IR 161/60 industrial robot show that it is also possible to apply the nonlinear flexible controller on industrial robots, and that, in comparison with classical rigid control, some improvements can be obtained.

In literature very few experimental results are found. The few experimental results found are obtained using idealised and specially developed test setups. The reason for this is that most control algorithms require an accurate model of the robot. Industrial robots are avoided because they are too far from ideal and therefore difficult to model. Other reasons are the problem of measurement or estimation of state variables, the dominant friction effects in commercial manipulators and the computational requirements of commercially available microprocessors which requires discrete time control.

The KUKA IR 161/60 industrial robot is considered as a robot with flexible joints. The flexibility is small. The flexibility of the links can be neglected. However the experiments show that even for this "stiff" robot, in comparison with rigid control, some improvements can be obtained: the tracking error is reduced by a factor 2, and the static deflection due to gravity is compensated.

1.4 Organisation of the dissertation

In this dissertation two main parts are distinguished, the first part describes the control design, while the second part gives simulation and experimental results.

Because the proposed controller is a model based controller, first, chapter 2, describes in a general way the derivation of the models.

A next logical step would be to describe, in a next chapter, the identification of the models. However, in this dissertation, identification is not treated in a general way in a separate chapter, but the techniques used are directly explained as they are applied for the identification of the different test set ups. This has the advantage that the techniques are directly illustrated by the examples.

Chapter 3 considers linear identification and control of a one flexible link robot. A state of the art of linear techniques is given. Most

attention is given to the techniques used in the experiments. In addition the linear controller is formulated as a special case of the proposed more general nonlinear controller.

Chapter 4 deals with nonlinear control of multi–link flexible robots. The proposed generalized nonlinear control method is formulated, and situated in the framework of the general nonlinear control theory developed by Freund.

Chapter 5 considers some practical implementation issues. This includes choice of sensors, estimation of velocities by a nonlinear state etimator or by differentiation, low pass filtering of measured signals and inclusion of the filter in the model, calculation of reference values for the state variables, the choice of the order of polynomials used in the trajectory generation, compensation of nonlinear Coulomb friction and the total control implementation.

The second part gives simulation and experimental results of the proposed controller.

Chapter 6 describes the experimental identification and control results of a flexible one–link robot. The identification and control design, as described in chapter 3, is used. Chapter 7 describes the simulation results of the control of a two–flexible link robot. Chapter 8 shows the experimental identification and control results of a test setup which is a model of a two–link robot with a flexible first joint. Chapter 9 shows the results on a KUKA IR 161/60 industrial robot. And finally in chapter 10 we draw our conclusions.

Three appendices give some additional information. The first appendix describes the general nonlinear decoupling control method as developed by Freund. Appendix B describes and compares the computed torque control solution, the singular perturbation solution and the generalized nonlinear decoupling solution for flexible joint robots. Appendix C treats trajectory generation and inverse kinematics.

References conclude the text.

Part I

Nonlinear Control of Flexible Robots: Theoretical Issues

Chapter 2

Modeling of flexible robots

2.1 Introduction

Essential for nonlinear control system design is the knowledge of an analytical model describing the manipulator dynamics. In this chapter, a dynamic model of robotic manipulators with compliant links and joints is developed based on a Lagrangian-assumed modes formulation. The final form of the equations is organized in a form similar to rigid manipulator equations.

For a general flexible robot, the detailed equations become quite complex, therefore the equations are only formulated in a general form. The manipulator is assumed to be planar. Extension to three dimensional manipulators is possible using the same formalism. The model is formulated in the tangent coordinate system. However other coordinate systems can be used which reduces the complexity of the model [11], [6]. This is shown, for a one–flexible link robot, in chapter 3. The derived equations for a two–flexible link robot moving in a vertical plane are given in chapter 7; equations for a two–link robot with flexible joints are given in chapters 8 and 9.

2.2 Flexible-arm kinematic description

The flexible–arm kinematic and dynamic description used in this chapter is based on the method by Cetinkunt and Book [14] and by Asada

[6].

Consider the kinematic chain shown in figure 2.1 representing a manipulator consisting of N flexible links.

Body fixed moving coordinates are assigned to the elements of the manipulator: Let O_i be the ith joint between link $i-1$ and link i. The endpoint of link i is denoted by O_i'. The coordinate axis $O_i x_i$ is directed along the tangent of the center line of link i in O_i. An additional coordinate axis $O_i' x_i'$ is directed along the tangent of the center line of link i in O_i'. The joint displacement θ_i, in the sequel also called rigid body coordinate, is defined as the angle between coordinate axis $O_{i-1}' x_{i-1}'$ and $O_i x_i$. $O_1 XY$ is the inertial coordinate frame. 2x2 homogeneous transformation matrices , $\mathbf{A}_i$, are used to describe the orientation of one coordinate frame with respect to another.

Since deformations in the longitudinal direction are assumed to be small, the distance between adjacent joints is invariant.

Let P_i be an arbitrary point on link i and $\mathbf{R}_i$ the position vector of point P_i with respect to the inertial reference frame (figure 2.2). The position of the same point is represented by $\mathbf{r}_i^i = [x_i \ , \ v_i]^T$, where the superscript means that the point is on link i, and the subscript indicates that the vector is represented with respect to the ith link coordinate system. The component x_i represents the longitudinal distance along the tangent, while the component v_i accounts for the deformation of the link beam. Note that the deformation is measured in the negative y_i axis, which is the convention in vibration theory.

From figure 2.2, it follows that

$$\mathbf{R}_i = \mathbf{R}_{0i} + \mathbf{C}_i \mathbf{r}_i^i$$

where

$$\mathbf{C}_i = \mathbf{A}_1 \mathbf{A}_2 ... \mathbf{A}_i \tag{2.1}$$

$$\mathbf{A}_i = \begin{bmatrix} cos\theta_i' & -sin\theta_i' \\ sin\theta_i' & cos\theta_i' \end{bmatrix}$$

θ_i' is defined as the angle between coordinate axis $O_{i-1} x_{i-1}$ and $O_i x_i$. Let $v_{i-1}(l_{i-1})$ be the deformation of the endpoint of link $i-1$, then

$$\theta_i' = \theta_i + \frac{\delta v_{i-1}(l_{i-1})}{\delta x_{i-1}} \tag{2.2}$$

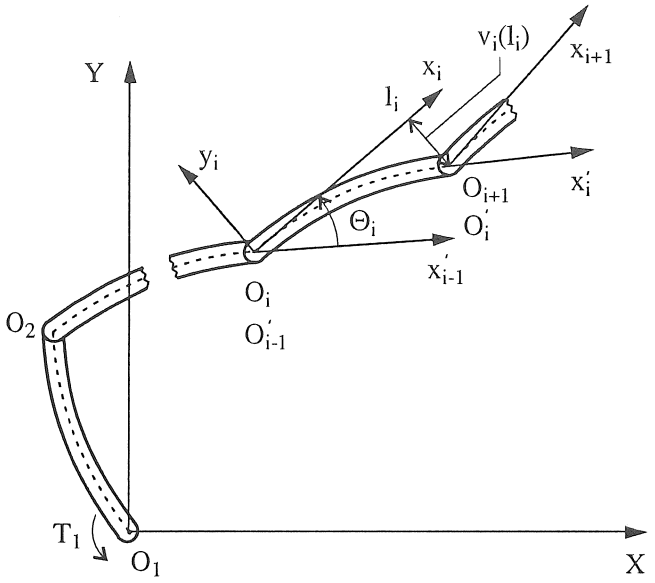

Figure 2.1: N-link flexible arm for planar motion

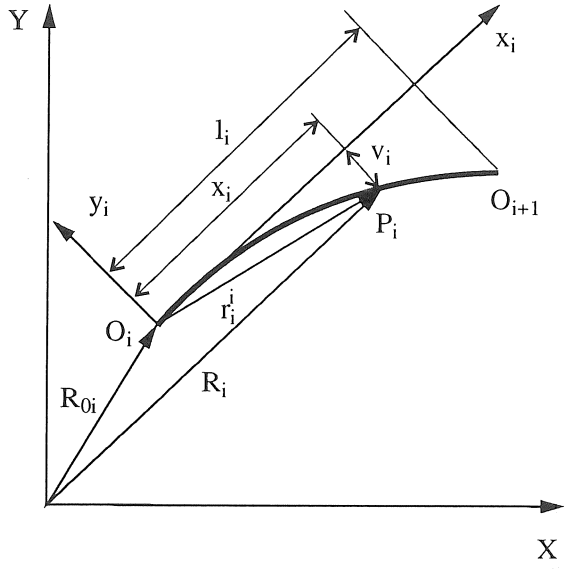

Figure 2.2: Link deformation represented in tangent coordinate system

The deflections are assumed small enough to justify the following approximation: $sin(\frac{\delta v_i(l_i)}{\delta x_i}) = \frac{\delta v_i(l_i)}{\delta x_i}$ and $cos(\frac{\delta v_i(l_i)}{\delta x_i}) = 1$. Then $\mathbf{A}_i$ is

$$\mathbf{A}_i = \begin{bmatrix} cos\theta_i & -sin\theta_i \\ sin\theta_i & cos\theta_i \end{bmatrix} + \begin{bmatrix} -sin\theta_i & -cos\theta_i \\ cos\theta_i & -sin\theta_i \end{bmatrix} \frac{\delta v_{i-1}(l_{i-1})}{\delta x_{i-1}}$$

Let $\mathbf{r}_k^i = [l_k \ , \ v_k]^T$, $(k < i)$ be the coordinates of joint $k+1$ viewed from the kth tangent coordinate system (l_k is the lenght of the kth link). Then position vector $\mathbf{R}_{0i}$ is given by

$$\mathbf{R}_{0i} = \sum_{k=1}^{i-1} \mathbf{C}_k \mathbf{r}_k^i$$

and

$$\mathbf{R}_i = \sum_{k=1}^{i} \mathbf{C}_k \mathbf{r}_k^i \tag{2.3}$$

The time derivative of $\mathbf{R}_i$ is then given by

$$\dot{\mathbf{R}}_i = \sum_{k=1}^{i} \dot{\mathbf{C}}_k \mathbf{r}_k + \mathbf{C}_k \dot{\mathbf{r}}_k \tag{2.4}$$

with

$$\dot{\mathbf{C}}_k = \sum_{j=1}^{i} (\frac{\delta \mathbf{C}_k}{\delta \theta_j'}) \dot{\theta}_j'$$

The deflections of the flexible links may be expressed as a weighted superposition of a finite set of spatial mode shapes that satisfy the geometric boundary conditions of the link. The lateral displacements of the flexible links are then written as

$$v_i(x_i, t) = \sum_{r=1}^{n_i} \Phi_{i,r}(x_i) q_{l\,i,r}(t) \tag{2.5}$$

where n_i is the number of modes used to describe the deflection of link i, $q_{l\,i,r}(t)$ are the time dependent modal amplitudes or weights, in the sequel also called generalized coordinates associated with the link flexibility and $\Phi_{i,r}(x_i)$ are the mode shape functions.
For the purpose of simulation, in the literature the assumed modes method is used ([8], [54], [40]). In order to generate an accurate model

with a minimal number of modes, appropriate component mode shapes are selected. The word "component" is used in the sense that these mode shapes are "building-blocks" for predicting the true system mode shapes.
They could be obtained from the analytical solution of an Euler-Bernoulli beam eigenfunction analysis. Or they could be obtained using a finite element analysis program. M. Oakley and R.H. Cannon [54] have studied theoretically and experimentally the selection of mode shapes for two–link flexible manipulators. For control design on real manipulators it is suitable to use experimentally determined natural frequencies and mode shapes.

2.3 Flexible–arm dynamics: Lagrangian formulation

Once the kinematic description of the system is set up, the next step in Lagrangian formulation is to form the kinetic and potential energies and take the derivatives with respect to θ_i and $q_{l\,i,r}$ to form the equations of motion:

$$\frac{d}{dt}\left(\frac{\delta E_K}{\delta \dot{\theta}_i}\right) - \frac{\delta E_K}{\delta \theta_i} + \frac{\delta E_P}{\delta \theta_i} = T_i \qquad i = 1, \ldots N \tag{2.6}$$

$$\frac{d}{dt}\left(\frac{\delta E_K}{\delta \dot{q}_{l\,i,r}}\right) - \frac{\delta E_K}{\delta q_{l\,i,r}} + \frac{\delta E_P}{\delta q_{l\,i,r}} = 0 \qquad i = 1, \ldots, N \qquad r = 1, \ldots, n_i \tag{2.7}$$

where $E_K = \sum_{i=1}^{N} E_{Ki}$: total kinetic energy
$E_P = \sum_{i=1}^{N} E_{Pi}$: total potential energy
T_i : ith motor torque

Assuming that each link is sufficiently thin, the kinetic energy of element i is given by:

$$E_{Ki} = \frac{1}{2}\int_o^{l_i} \rho_i S_i \dot{\mathbf{R}}_i^T \dot{\mathbf{R}}_i dx_i \tag{2.8}$$

with ρ_i : density of the ith link
S_i : cross section area of the ith link

The total potential energy of the system is the sum of the gravitational potential energy, E_{Pg} , and the elastic potential energy E_{Pe}.

$$E_P = E_{Pg} + E_{Pe}$$

The gravitational potential energy of element i, E_{Pgi} , is given by

$$E_{Pgi} = \int_0^{l_i} \rho_i S_i \mathbf{g}^T (\sum_{k=1}^{i} \mathbf{C}_k \mathbf{r}_k^i) dx_i \tag{2.9}$$

where $\mathbf{g}^T = [g_x \;\; g_y]$ contains the components of the gravitation vector. Let $(EI)_i$ be the bending stiffness of link i, then the elastic potential energy stored in link i is given by

$$E_{Pei} = \frac{1}{2} \int_o^{l_i} (EI)_i \left[\frac{\delta^2 v_i}{\delta x_i^2} \right]^2 dx_i \tag{2.10}$$

Noting the truncated modal approximations for the deformation coordinates of the links:

$$E_{Pei} = \sum_{j=1}^{n_i} \frac{1}{2} q_{l\,i,j}^2 \int_o^{l_i} (EI)_i \left[\frac{\delta^2 \Phi_{i,j}}{\delta x_i^2} \right]^2 dx_i = \sum_{j=1}^{n_i} \frac{1}{2} k_{i,j} q_{l\,i,j}^2$$

with

$$k_{i,j} = \int_o^{l_i} (EI)_i \left[\frac{\delta^2 \Phi_{i,j}}{\delta x_i^2} \right]^2 dx_i$$

2.4 Inclusion of flexible joint dynamics

Until now the joints are considered rigid. Inclusion of flexible joint dynamics into the model involves ([14]):

1. redefining the mass distribution of the links: the rotor/gear assembly inertia, I_{mi}, (reduced to link axis) is separated from the inertia of the links.

2. adding a set of second order equations to equation 2.6 as a result of joint flexibility and inertia.

Two assumptions are made:

- The first assumption is that the rotational kinetic energy of each joint about its own center of mass is only due to its own rotation. Rotational kinetic energy due to rotation of previous joints and links is neglected. This amounts to neglecting terms in the order of gear reduction ratio, which is typically in the order of 1:100. Translational kinetic energy due to both previous joints and elastic deformations is taken into account.
- A second assumption is that rotor/gear assembly inertia is symmetric about the rotor axis of rotation such that gravitational potential energy, and translational velocity of the joint center of mass are independent of rotor position. This assumption is generally satisfied by joint assemblies of most industrial robots.

Elastic mechanical coupling between a joint and a link is modeled as a torsional spring: the relative position between the joint rotor and an elastically coupled link, $(\theta_{li} - \theta_{mi}) = q_{ji}$, corresponds to a force

$$K_{ji}(\theta_{li} - \theta_{mi})$$

with K_{ji} the spring constant of joint i, θ_{li} the ith link position (joint angle after the elastic transmission), and θ_{mi} the ith motor position (joint angle before the elastic transmission) reduced to link axis.

As a result equation 2.6 is modified to the following form:

$$\frac{d}{dt}(\frac{\delta E_K}{\delta \dot{\theta}_{li}}) - \frac{\delta E_K}{\delta \theta_{li}} + \frac{\delta E_P}{\delta \theta_{li}} = -K_{ji}(\theta_{li} - \theta_{mi}) \quad i = 1, ...N \tag{2.11}$$

The additional equation is given by:

$$I_{mi}\ddot{\theta}_{mi} - K_{ji}(\theta_{li} - \theta_{mi}) = T_i \tag{2.12}$$

2.5 Dynamic model

For control design the model (eq. 2.6 and 2.7) is put in compact form:

$$\begin{bmatrix} \mathbf{M}_r & \mathbf{M}_{rfl} \\ \mathbf{M}_{rfl}^T & \mathbf{M}_{fl} \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{\theta}}_l \\ \ddot{\mathbf{q}}_l \end{bmatrix} + \begin{bmatrix} \mathbf{C}_r(\boldsymbol{\theta}_l, \mathbf{q}_l, \dot{\boldsymbol{\theta}}_l, \dot{\mathbf{q}}_l) \\ \mathbf{C}_{fl}(\boldsymbol{\theta}_l, \mathbf{q}_l, \dot{\boldsymbol{\theta}}_l, \dot{\mathbf{q}}_l) \end{bmatrix} + \begin{bmatrix} \mathbf{G}_r(\boldsymbol{\theta}_l, \mathbf{q}_l) \\ \mathbf{G}_{fl}(\boldsymbol{\theta}_l, \mathbf{q}_l) \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_{fl} \end{bmatrix} \begin{bmatrix} \boldsymbol{\theta}_l \\ \mathbf{q}_l \end{bmatrix} = \begin{bmatrix} \mathbf{T} \\ \mathbf{0} \end{bmatrix} \tag{2.13}$$

or

$$\mathbf{M}(\theta_l)\begin{bmatrix}\ddot{\theta}_l\\ \ddot{\mathbf{q}}_l\end{bmatrix}+\mathbf{C}(\theta_l,\mathbf{q}_l,\dot{\theta}_l,\dot{\mathbf{q}}_l)+\mathbf{G}(\theta_l,\mathbf{q}_l)+\mathbf{K}\begin{bmatrix}\theta_l\\ \mathbf{q}_l\end{bmatrix}=\begin{bmatrix}\mathbf{T}\\ \mathbf{0}\end{bmatrix} \quad (2.14)$$

where, for a manipulator with N rigid degrees of freedom,
$\theta_l = [\theta_{l1}\ \theta_{l2}\ ...\theta_{lN}]^T$ is the vector of link positions,
$\mathbf{q}_l = [q_{l\,1,1}\ q_{l\,1,2}\ \cdots q_{l\,1,n_1}\ q_{l\,2,1}\ q_{l\,2,2}\ \cdots q_{l\,2,n_2}\ \cdots\ q_{l\,N,1}\ q_{l\,N,2}\ \cdots q_{l\,N,n_N}]^T$ is the vector of generalized coordinates associated with the link flexibilities, $\mathbf{T}$ is the vector of motor torques, $\mathbf{M}(\theta_l)$ is the generalized inertia matrix where the dependency to flexible coordinates is neglected, $\mathbf{C}(\theta_l,\mathbf{q}_l,\dot{\theta}_l,\dot{\mathbf{q}}_l)$ is a vector containing the nonlinear centrifugal and coriolis terms, $\mathbf{G}(\theta_l,\mathbf{q}_l)$ is the gravity vector and $\mathbf{K}_{fl} = Diag\{k_{i,j}\}$ is the modal generalized stiffness matrix.
The mass matrix, $\mathbf{M}(\theta_l)$, is symmetric and positive definite.

The terms describing centrifugal and coriolis forces, gravitational forces and modal stiffness can be summarized by one nonlinear function:

$$\begin{bmatrix}\mathbf{F}_r(\theta_l,\mathbf{q}_l,\dot{\theta}_l,\dot{\mathbf{q}}_l)\\ \mathbf{F}_{fl}(\theta_l,\mathbf{q}_l,\dot{\theta}_l,\dot{\mathbf{q}}_l)\end{bmatrix}=\begin{bmatrix}\mathbf{C}_r(\theta_l,\mathbf{q}_l,\dot{\theta}_l,\dot{\mathbf{q}}_l)\\ \mathbf{C}_{fl}(\theta_l,\mathbf{q}_l,\dot{\theta}_l,\dot{\mathbf{q}}_l)\end{bmatrix}+\begin{bmatrix}\mathbf{G}_r(\theta_l,\mathbf{q}_l)\\ \mathbf{G}_{fl}(\theta_l,\mathbf{q}_l)\end{bmatrix}+\begin{bmatrix}\mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{K}_{fl}\end{bmatrix}\begin{bmatrix}\theta_l\\ \mathbf{q}_l\end{bmatrix} \quad (2.15)$$

Generally, a term describing the structural damping ,

$$\begin{bmatrix}\mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{D}_{fl}\end{bmatrix}\begin{bmatrix}\dot{\theta}_l\\ \dot{\mathbf{q}}_l\end{bmatrix}$$

and a term describing the motor friction

$$\begin{bmatrix}\mathbf{T}_f\\ \mathbf{0}\end{bmatrix}$$

can be added to this nonlinear function 2.15.
The model becomes:

$$\begin{bmatrix}\mathbf{M}_r(\theta_l) & \mathbf{M}_{rfl}(\theta_l)\\ \mathbf{M}_{rfl}^T(\theta_l) & \mathbf{M}_{fl}(\theta_l)\end{bmatrix}\begin{bmatrix}\ddot{\theta}_l\\ \ddot{\mathbf{q}}_l\end{bmatrix}+\begin{bmatrix}\mathbf{F}_r(\theta_l,\mathbf{q}_l,\dot{\theta}_l,\dot{\mathbf{q}}_l)\\ \mathbf{F}_{fl}(\theta_l,\mathbf{q}_l,\dot{\theta}_l,\dot{\mathbf{q}}_l)\end{bmatrix}=\begin{bmatrix}\mathbf{T}\\ \mathbf{0}\end{bmatrix} \quad (2.16)$$

The dynamic model of a flexible joint robot is given by:

$$\begin{bmatrix} \mathbf{M}_r - \mathbf{J} & \mathbf{M}_{rfl} \\ \mathbf{M}_{rfl}^T & \mathbf{M}_{fl} \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{\theta}}_l \\ \ddot{\mathbf{q}}_l \end{bmatrix} + \begin{bmatrix} \mathbf{F}_r(\boldsymbol{\theta}_l, \mathbf{q}_l, \dot{\boldsymbol{\theta}}_l, \dot{\mathbf{q}}_l) \\ \mathbf{F}_{fl}(\boldsymbol{\theta}_l, \mathbf{q}_l, \dot{\boldsymbol{\theta}}_l, \dot{\mathbf{q}}_l) \end{bmatrix} = \begin{bmatrix} -\mathbf{K}_j \mathbf{q}_j \\ \mathbf{0} \end{bmatrix} \tag{2.17}$$

and

$$\mathbf{J}\ddot{\boldsymbol{\theta}}_m - \mathbf{K}_j \mathbf{q}_j = \mathbf{T} \tag{2.18}$$

where $\mathbf{J} = Diag\{I_{mi}\}$
$\mathbf{K}_j = Diag\{K_{ji}\}$
$\boldsymbol{\theta}_m = [\theta_{m1}\ \theta_{m2}\ ...\theta_{mN}]^T$ is the vector of motor positions.
$\mathbf{q}_j = [q_{j1}\ q_{j2}\ ...q_{jN}]^T$ is the vector of generalized coordinates associated with the joint flexibility

Writing the dynamic model (2.17 and 2.18) in function of the vector of motor positions, $\boldsymbol{\theta}_m$, the vector of generalized coordinates associated with the joint flexibility, $\mathbf{q}_j$, and the vector of generalized coordinates associated with the link flexibility, $\mathbf{q}_l$, (substituting $\boldsymbol{\theta}_l$ by $\mathbf{q}_j + \boldsymbol{\theta}_m$) gives:

$$\begin{bmatrix} \mathbf{M}_r - \mathbf{J} & \mathbf{M}_r - \mathbf{J} & \mathbf{M}_{rfl} \\ \mathbf{M}_{rfl}^T & \mathbf{M}_{rfl}^T & \mathbf{M}_{fl} \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{\theta}}_m \\ \ddot{\mathbf{q}}_j \\ \ddot{\mathbf{q}}_l \end{bmatrix} + \begin{bmatrix} \mathbf{F}_r \\ \mathbf{F}_{fl} \end{bmatrix} = \begin{bmatrix} -\mathbf{K}_j \mathbf{q}_j \\ \mathbf{0} \end{bmatrix} \tag{2.19}$$

and

$$\mathbf{J}\ddot{\boldsymbol{\theta}}_m - \mathbf{K}_j \mathbf{q}_j = \mathbf{T} \tag{2.20}$$

Adding the first equation of 2.19 and equation 2.20 gives:

$$\mathbf{M}_r \ddot{\boldsymbol{\theta}}_m + (\mathbf{M}_r - \mathbf{J})\, \ddot{\mathbf{q}}_j + \mathbf{M}_{rfl}\, \ddot{\mathbf{q}}_l + \mathbf{F}_r = \mathbf{T}$$

The dynamic model of a flexible joint robot can thus be written as:

$$\begin{bmatrix} \mathbf{M}_r & \mathbf{M}_r - \mathbf{J} & \mathbf{M}_{rfl} \\ \mathbf{M}_r - \mathbf{J} & \mathbf{M}_r - \mathbf{J} & \mathbf{M}_{rfl} \\ \mathbf{M}_{rfl}^T & \mathbf{M}_{rfl}^T & \mathbf{M}_{fl} \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{\theta}}_m \\ \ddot{\mathbf{q}}_j \\ \ddot{\mathbf{q}}_l \end{bmatrix} + \begin{bmatrix} \mathbf{F}_r \\ \mathbf{F}_r + \mathbf{K}_j \mathbf{q}_j \\ \mathbf{F}_{fl} \end{bmatrix} = \begin{bmatrix} \mathbf{T} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \tag{2.21}$$

The dynamic model of a flexible link and of a flexible joint robot (eqn. 2.16 and 2.21) can both be formulated in the general form:

$$\begin{bmatrix} \mathbf{M}_{11} & \mathbf{M}_{12} \\ \mathbf{M}_{12}^T & \mathbf{M}_{22} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}}_0 \\ \ddot{\mathbf{q}}_f \end{bmatrix} + \begin{bmatrix} \mathbf{F}_{\mathbf{q}_0}(\mathbf{q}_0, \mathbf{q}_f, \dot{\mathbf{q}}_0, \dot{\mathbf{q}}_f) \\ \mathbf{F}_{\mathbf{q}_f}(\mathbf{q}_0, \mathbf{q}_f, \dot{\mathbf{q}}_0, \dot{\mathbf{q}}_f) \end{bmatrix} = \begin{bmatrix} \mathbf{T} \\ \mathbf{0} \end{bmatrix} \tag{2.22}$$

where for a rigid joint robot, the rigid body coordinate $\mathbf{q}_0$ is equal to $\theta_l = \theta_m$, the flexible coordinate $\mathbf{q}_f$ is equal to $\mathbf{q}_l$, and

$$\begin{aligned} \mathbf{M}_{11} &= \mathbf{M}_r, \\ \mathbf{M}_{12} &= \mathbf{M}_{rfl}, \\ \mathbf{M}_{22} &= \mathbf{M}_{fl}, \\ \mathbf{F}_{\mathbf{q}_0} &= \mathbf{F}_r, \\ \mathbf{F}_{\mathbf{q}_f} &= \mathbf{F}_{fl}. \end{aligned} \tag{2.23}$$

For a flexible joint robot, the rigid body coordinate $\mathbf{q}_0$ is equal to θ_m, the flexible coordinate $\mathbf{q}_f$ is equal to $[\mathbf{q}_j \quad \mathbf{q}_l]^T$, and

$$\begin{aligned} \mathbf{M}_{11} &= \mathbf{M}_r, \\ \mathbf{M}_{12} &= [\, \mathbf{M}_r - \mathbf{J} \quad \mathbf{M}_{rfl} \,], \\ \mathbf{M}_{22} &= \begin{bmatrix} \mathbf{M}_r - \mathbf{J} & \mathbf{M}_{rfl} \\ \mathbf{M}_{rfl}^T & \mathbf{M}_{fl} \end{bmatrix}, \\ \mathbf{F}_{\mathbf{q}_0} &= \mathbf{F}_r, \\ \mathbf{F}_{\mathbf{q}_f} &= \begin{bmatrix} \mathbf{F}_r + \mathbf{K}_j \mathbf{q}_j \\ \mathbf{F}_{fl} \end{bmatrix}. \end{aligned} \tag{2.24}$$

This general form of the dynamic model of a flexible robot will be used in nonlinear control design (chapter 4).

2.6 Conclusion

In this chapter a dynamic model of robotic manipulators with compliant links and joints is developed based on a Lagrangian formulation. The equations are formulated in a general form. This model is written out explicitely for a one–flexible link robot in chapter 3, for a two–flexible link robot in chapter 7, for a two–link robot with flexibility in the first joint in chapter 8 and for a two–link robot with flexibility in the two joints and with gear transmissions in chapter 9. Because of the high gear transmission rates, in chapter 9, cross–coupling terms due to the motor mass are neglected. However, the two–link robot with flexibility in the first joint, described in chapter 8, uses direct drive

motors. In this model the cross–coupling terms due to the motor mass are not neglected.

The final form of the equations is organized in a form similar to rigid manipulator equations. This model forms the basis of the non-linear control system design described in chapter 4.

Chapter 3

Linear identification and control of one–link flexible robots

3.1 Introduction

Identification and control of a flexible one–link robot is a first step in the development of a controller for a more complex configuration. The model describing the dynamics of a flexible one–link robot moving in the horizontal plane is linear. Classical linear identification and control techniques can be used.

This chapter gives an overview of concepts used in linear identification and control. Most attention is given to the concepts which are used in the experiments on control of a one–flexible link robot (chapter 6).

Linear strategies, applied to the flexible one–link robot, must be supplemented with nonlinear techniques for the more complex case. Nonlinear control design will be described in chapter 4. The linear control design, described in this chapter, is formulated in the same way as the more general nonlinear control design.

A state feedback controller with acceleration feedforward is discussed.

Section 3.2 derives the linear physical model of the flexible one–link robot, based on the Lagrangian formulation, as described in chapter 2.

Section 3.3 describes the least squares identification of linear models, based on the identification of transfer functions.
Because of the linearity of the model, for state feedback control design it is not necessary to transform the identified model to its physical equivalent. Therefore the state feedback control design, described in section 3.4, is based on an arbitrary discrete time model, without reference to the physical background.
Observation spill-over is avoided by filtering the output signals using a second order digital filter. These filters extend the model order.
Acceleration feedforward is introduced in the state feedback controller. Together with an appropriate calculation of the desired states, this feedforward reduces tracking errors to negligible values. The calculation of the acceleration feedforward is based on the quasi–static behaviour of the system (only rigid body dynamics are considered).
Section 3.5 describes the limitations of linear control techniques for flexible robots with nonlinear joint friction. Integral control eliminates positioning errors caused by the nonlinear joint friction. Limit cycles are avoided by choosing a limited closed loop bandwidth. Also a strategy is proposed to reduce limit cycles by phasing out the feedback once the final position is reached.

3.2 Model of a one flexible link robot moving in the horizontal plane

Figure 3.1 gives the definition of the one flexible link robot moving in the horizontal plane. Table 3.1 gives a list of used symbols. The equations of motion in the tangent coordinate frame are derived from the Lagrangian–assumed modes formulation, as described in chapter 2.

They are summarized by (equation 2.22)

$$\begin{bmatrix} \mathbf{M}_{11}(\theta) & \mathbf{M}_{12}(\theta) \\ \mathbf{M}_{12}^T(\theta) & \mathbf{M}_{22}(\theta) \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \ddot{\mathbf{q}} \end{bmatrix} + \begin{bmatrix} \mathbf{F}_\theta(\theta, \mathbf{q}, \dot{\theta}, \dot{\mathbf{q}}) \\ \mathbf{F}_\mathbf{q}(\theta, \mathbf{q}, \dot{\theta}, \dot{\mathbf{q}}) \end{bmatrix} = \begin{bmatrix} T \\ 0 \end{bmatrix} \tag{3.1}$$

with

$$\begin{aligned} \mathbf{q} &= [q_1 \ \dots \ q_n]^T \\ \mathbf{F}_\theta &= 0 \\ \mathbf{F}_\mathbf{q} &= [F_{q_1} \ \dots \ F_{q_n}]^T \end{aligned}$$

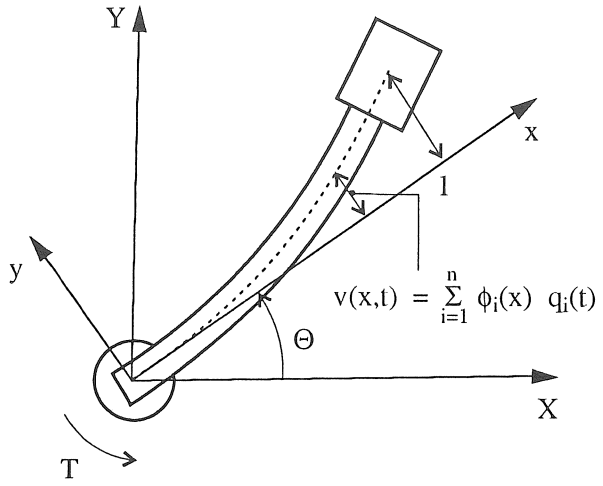

Figure 3.1: One flexible link model

I_t	$= I_b + I_h + I_p + m_p l^2$: total inertia of beam, hub and payload around z through O
l	length of the beam
m_p, I_p	mass of the payload and moment of inertia around its center of gravity
$\Phi_i(x)$	constrained shape functions
$q_i(t)$	generalized flexible coordinates
K_i	$= \int_0^l \rho S x \Phi_i(x) dx$
γ_i	$= K_i + m_p l \Phi_i(l) + I_p \frac{d\Phi_i}{dx}(l)$
m_i	$= \int_0^l \rho S \Phi_i^2(x) dx$: generalized mass of the ith mode
m_i'	$= m_i + I_p \left[\frac{\delta \Phi_i}{\delta x}(l)\right]^2 + m_p \Phi_i^2(l)$
$m_{i,j}$	$= \int_0^l \rho S \Phi_i(x) \Phi_j(x) dx$
k_i	$= \int_o^l (EI) \left[\frac{\delta^2 \Phi_i}{\delta x^2}\right]^2 dx = m_i' \omega_i^2$
ω_i	ith cantilevered mode eigenpulsation
ζ_i	modal damping of the ith mode

Table 3.1: Nomenclature

$$
\begin{aligned}
F_{q_i} &= m_i'\omega_i^2\ q_i + 2m_i'\omega_i\zeta_i\ \dot{q}_i \qquad\qquad i = 1, ..., n \\
\mathrm{M}_{11} &= I_t \\
\mathrm{M}_{12}^{1j} &= \gamma_j \qquad\qquad j = 1, ..., n \\
\mathrm{M}_{22}^{ij} &= m_{i,j} + I_p \frac{d\Phi_i}{dx}(l)\frac{d\Phi_j}{dx}(l) + m_p\Phi_i(l)\Phi_j(l)
\end{aligned}
$$

where i : row index
j : column index
n : number of modes

The tip rotation (in arc length) is given by

$$y_e(l,t) = l\theta(t) + \sum_{i=1}^{n} \Phi_i(l)q_i(t) \tag{3.2}$$

An alternative formulation of the model is developped by Cannon and Schmitz [11]. The equations are simplified by employing a system of orthogonal functions for the mode shape functions $\Phi_i(x)$, denoted by $\Psi_i(x'')$, namely

$$\int_0^l \Psi_i(x'')\Psi_j(x'')dm = \delta_{ij}I_t$$

which can be written as

$$\int_0^l \rho S\Psi_i(x'')\Psi_j(x'')dx + I_p\frac{d\Psi_i}{dx''}(l)\frac{d\Psi_j}{dx''}(l) + m_p\Psi_i(l)\Psi_j(l) = \delta_{ij}I_t$$

In addition a rotation angle α is defined so that the rigid body motion $x''\alpha(t)$ can be considered as a zero frequency mode orthogonal to the flexible modes. The new body fixed moving coordinate system is denoted by $0x''y''$. Figure 3.2 represents the one–flexible link robot in the two coordinate systems $0xy$ and $0x''y''$.

The parameters in model 3.1 become

$$
\begin{aligned}
\mathrm{M}_{22}^{ij} &= \delta_{ij}I_t \\
\gamma_j &= 0 \\
m_i' &= I_t
\end{aligned}
\tag{3.3}
$$

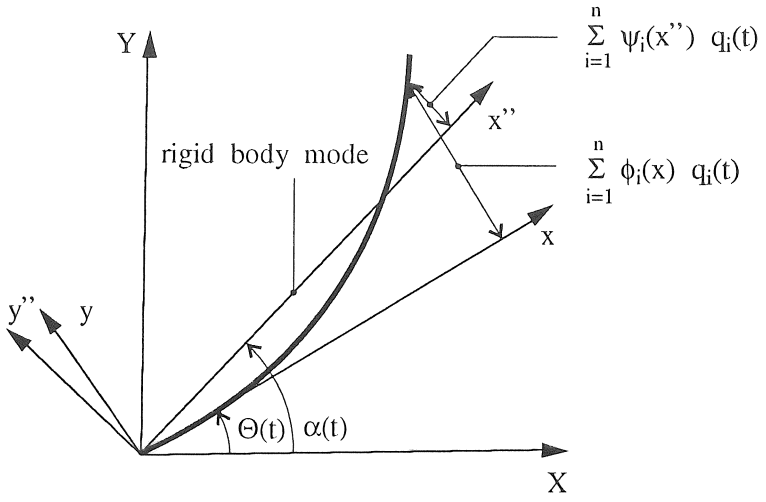

Figure 3.2: One flexible link robot: link deformation in the tangent coordinate system Oxy and in the coordinate system $0x''y''$.

The model becomes

$$I_t\ddot{\alpha} = T$$
$$I_t\ddot{q}_i + I_t\omega_i^2\, q_i + 2I_t\omega_i\zeta_i\, \dot{q}_i = \frac{d\Psi_i}{dx''}(0)T \qquad i = 1, ..., n \quad (3.4)$$

The total rotation angle θ is given by

$$\theta = \alpha + \sum_{i=1}^{n} \frac{d\Psi_i}{dx''}(0)q_i \quad (3.5)$$

The tip rotation is given by

$$y_e = l\alpha + \sum_{i=1}^{n} \Psi_i(l)q_i \quad (3.6)$$

Note the term $\frac{d\Psi_i}{dx''}(0)T$ in the right hand side of equation 3.4. While in the tangential coordinate frame the generalized force

$$Q_i = \frac{\delta W}{\delta q_i}$$

with

$$W = T\theta = T\ (\ \alpha + \sum_{i=1}^{n} \frac{d\Psi_i}{dx''}(0)\ q_i\ )$$

is equal to zero, in this "orthogonal" coordinate frame the generalized force Q_i is equal to $\frac{d\Psi_i}{dx''}(0)T$.

The obtained models 3.1 and 3.4 are linear. The state space representation for model 3.4 is given by

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}T \\ y_e &= \mathbf{C}\mathbf{x} \end{aligned} \tag{3.7}$$

where

$$\mathbf{x} = [\alpha\ \ \dot{\alpha}\ \ q_1\ \ \dot{q}_1\ \ldots\ q_n\ \ \dot{q}_n]^T$$

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & & & & & & \\ 0 & 0 & & & & & & \\ & & 0 & 1 & & & & \\ & & -\omega_1^2 & -2\zeta_1\omega_1 & & & & \\ & & & & \cdot & & & \\ & & & & & \cdot & & \\ & & & & & & \cdot & \\ & & & & & & 0 & 1 \\ & & & & & & -\omega_n^2 & -2\zeta_n\omega_n \end{bmatrix}$$

$$\mathbf{B} = \frac{1}{I_t} \begin{bmatrix} 0 \\ 1 \\ 0 \\ \frac{d\Psi_1}{dx''}(0) \\ \cdot \\ \cdot \\ \cdot \\ 0 \\ \frac{d\Psi_n}{dx''}(0) \end{bmatrix}$$

$$\mathbf{C} = [l\ \ 0\ \ \Psi_1(l)\ \ 0\ \ \ldots\ \ \Psi_n(l)\ \ 0]$$

3.3 Least squares identification

In the previous section 3.2, the model of a one flexible link robot is obtained from a physical analysis of the system. However, because the obtained model is linear, for linear control design standard linear models may be employed, without reference to the physical background (black box approach).

A good overview of linear system identification techniques is found in [67]. This section gives a short overview of the most important methods used in this dissertation.

The identification of the state space model is based on the identification of the corresponding transfer functions . For control design discrete time models are needed. Therefore it is convenient to identify the transfer functions directly in discrete time. In general, the discrete transfer function that relates the z-transform of the input $U(z^{-1})$ to the z-transform of the output $Y(z^{-1})$ has the following form:

$$\begin{aligned} H(z^{-1}) = \frac{Y(z^{-1})}{U(z^{-1})} &= \frac{B(z^{-1})}{A(z^{-1})} \\ &= \frac{b_0 + b_1 z^{-1} + ... + b_{n_b} z^{-n_b}}{1 + a_1 z^{-1} + ... + a_{n_a} z^{-n_a}} \end{aligned} \tag{3.8}$$

The identification of these transfer functions proceeds either in the time domain (subsection 3.3.1) or in the frequency domain (subsection 3.3.2).

An example of a standard discrete linear state space model derived from the identified transfer function 3.8 is the model in controller canonical form

$$\mathbf{x}(k+1) = \begin{bmatrix} -a_1 & -a_2 & . & . & . & -a_{n_a} \\ 1 & 0 & & & & \\ 0 & 1 & & & & \\ & & . & & & \\ & & & . & & \\ & & & & . & \\ & & & & 1 & 0 \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} 1 \\ 0 \\ 0 \\ . \\ . \\ . \\ 0 \end{bmatrix} T$$

$$y(k) = [b_1 - b_0 a_1 \ldots b_{n_b} - b_o a_{nb} \quad - b_0 a_{n_b+1} \ldots - b_0 a_n]\, \mathbf{x}(k) + b_0 T$$

The physical model is used to determine the complexity of the discrete transfer function (the order of the numerator n_b and denominator n_a).

In chapter 6, " Experimental results on a flexible one link robot", the procedure described here, is used to identify a standard input output model, without identifying the unknown physical parameters. However if necessary, physical parameters can be obtained by comparing the standard numerical transfer functions with the analytical expressions of the transfer functions. This is particularly useful for the identification of nonlinear systems, where, from the identification of the linearized model in a certain position, some physical parameters can be derived (see chapter 9, " Experimental results on a KUKA IR 161/60 industrial robot ").

3.3.1 Least squares time domain identification

A least squares parameter estimation algorithm calculates the unknown parameters of the transfer functions. These estimations are based on measured input-output signals which result from an excitation with a band limited random signal. The use of a band limited random signal avoids the unnecessary excitation of the higher modes during the measurement.

The unknown parameters of the transfer function, are calculated using a least squares time domain parameter estimation algorithm which fits the parameters of a difference equation to a given set of input–output data.

Low pass filtering is necessary in order to obtain reliable estimates. It is well known [50] that least squares methods lead to biased solutions if the input-output measurements are corrupted with measurement noise, high order oscillations and harmonics caused by nonlinearities, etc ... The solution to this problem is low pass filtering of the data. In the experiments, a second order digital low pass Butterworth filter has been used for this purpose. The cut-off frequency of the Butterworth filter has been chosen a little higher than the highest resonance frequency to be modeled.

The least squares time domain parameter estimator can be formulated as follows: The discrete transfer function that relates the z-transform of the input $U(z^{-1})$ to the z-transform of the output $Y(z^{-1})$

has the following form:

$$\frac{Y(z^{-1})}{U(z^{-1})} = \frac{B(z^{-1})}{A(z^{-1})} = \frac{b_0 + b_1 z^{-1} + ... + b_{n_b} z^{-n_b}}{1 + a_1 z^{-1} + ... + a_{n_a} z^{-n_a}} \tag{3.9}$$

This can be reformulated as follows:

$$(1 + a_1 z^{-1} + ... + a_{n_a} z^{-n_a}) Y(z^{-1}) = (b_0 + b_1 z^{-1} + ... + b_{n_b} z^{-n_b}) U(z^{-1}) \tag{3.10}$$

In the time domain this corresponds to a difference equation:

$$y(k) + a_1 y(k-1) + ... + a_{n_a} y(k-n_a) = b_0 u(k) + ... + b_{n_b} u(k-n_b) \tag{3.11}$$

or:

$$y(k) = -a_1 y(k-1) - ... - a_{n_a} y(k-n_a) + b_0 u(k) + ... + b_{n_b} u(k-n_b) \tag{3.12}$$

If we introduce the regression vector $\psi(k)$ and the parameter vector θ_0 we get:

$$y(k) = \psi^T(k) \theta_0 \tag{3.13}$$

with

$$\begin{aligned} \psi(k) &= \left[\; -y(k-1) \ldots -y(k-n_a) \;\; u(k) \ldots u(k-n_b) \; \right]^T \\ \theta_0 &= \left[\; a_1 \; \ldots a_{n_a} \;\; b_0 \ldots b_{n_b} \; \right]^T \end{aligned}$$

Given a set of input-output data we have a set of such equations for each value of k. This overdetermined set of equations can be solved in a least squares sense ([24], [33], [57]) to obtain the unknown parameter vector, θ_0 .

A priori information can be taken into account during the parameter estimation by putting it in the left hand side of equation 3.13.

3.3.2 Frequency domain identification based on a stepped sine measured frequency response function

Frequency domain identification techniques estimate parameters of postulated transfer functions models from measured frequency response functions that relate inputs to outputs in a certain frequency range. The measured frequency response functions are obtained using pure sinusoidal excitation (stepped sine excitation).

The most important advantages of the stepped sine excitation technique compared to other techniques (swept sine, white noise, hammer impact etc...) are a high signal to noise ratio, and the possibility to use a variable distribution of frequency points such that interesting areas (e.g. resonance and anti resonance frequencies) are measured with a higher frequency resolution.

The steady state response of a linear system to a sinusoidal input at a particular pulsation ω_i, $u_{\omega_i}(t)$, is a sinusoid with the same frequency but a different amplitude and phase, $y_{\omega_i}(t)$. A Fourier analysis of these signals results in one nonzero component for each signal. For the input, $u_{\omega_i}(t)$, this component is represented by $a^u_{\omega_i} + jb^u_{\omega_i}$. For the output, $y_{\omega_i}(t)$, this component is represented by $a^y_{\omega_i} + jb^y_{\omega_i}$. The complex value of the measured frequency response function at that particular pulsation ω_i is the ratio of these two complex numbers. In [67] a method is explained to calculate, at every particular frequency, $a^u_{\omega_i} + jb^u_{\omega_i}$ and $a^y_{\omega_i} + jb^y_{\omega_i}$ from the measured time domain input–output data.

The frequency domain identification method starts from the discrete time transfer function description of the linear single-input single-output system (equation 3.8). The parameters of $H(z^{-1})$ are estimated by comparing $H(z^{-1})$, evaluated for $z = e^{j\omega_i T_s}$, with the values of the measured frequency response function at different pulsations ω_i:

$$H(e^{-j\omega_i T_s}) = \frac{a^y_{\omega_i} + jb^y_{\omega_i}}{a^u_{\omega_i} + jb^u_{\omega_i}} \tag{3.14}$$

T_s is the sampling period of the discrete time model. This makes it possible to formulate the parameter estimation algorithm as a set of homogeneous equations that are solved in a total least squares sense. Combining equation 3.8 and 3.14 results in the following complex linear relation:

$$\begin{aligned} 0 &= (a_0 + a_1(e^{-j\omega_i T_s}) + ... + a_{n_a}(e^{-j\omega_i n_a T_s}))(a^y_{\omega_i} + jb^y_{\omega_i}) \\ &- (b_0 + b_1(e^{-j\omega_i T_s}) + ... + b_{n_b}(e^{-j\omega_i n_b T_s}))(a^u_{\omega_i} + jb^u_{\omega_i}) \end{aligned} \tag{3.15}$$

Both the real and the imaginary part of this equation have to be zero.

This leads to the following two linear equations:

$$\begin{bmatrix} a^y_{\omega_i} & \Re([a^y_{\omega_i} + jb^y_{\omega_i}]e^{-j\omega_i T_s}) & \dots & -\Re(a^u_{\omega_i} + jb^u_{\omega_i}) & \dots \\ b^y_{\omega_i} & \Im([a^y_{\omega_i} + jb^y_{\omega_i}]e^{-j\omega_i T_s}) & \dots & -\Im(a^u_{\omega_i} + jb^u_{\omega_i}) & \dots \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ . \\ . \\ . \\ a_{n_a} \\ b_0 \\ b_1 \\ . \\ . \\ . \\ b_{n_b} \end{bmatrix} = 0 \tag{3.16}$$

where $\Re$ denotes the real part of a complex number and $\Im$ denotes the imaginary part of a complex number. Given the frequency response function measured at k different frequencies ω_i, i=1,...,k, we have a overdetermined set of homogeneous equations that can be solved in a total least squares sense ([33], [78], [1]) to obtain the unknown parameter vector

$$\begin{bmatrix} a_0 & a_1 & \dots a_{n_a} & b_0 & b_1 \dots b_{n_b} \end{bmatrix}^T \tag{3.17}$$

Note that here the first coefficient of the denominator polynomial $A(z^{-1})$ is a_0 instead of 1. The solution of this set of homogeneous equations can always be scaled such that a_0 equals 1.

3.3.3 Spill-over

A one-link flexible robot has an infinite number of resonance frequencies and anti-resonance frequencies. The identified model and the controller considers only the first n resonance and anti–resonance frequencies. The unmodelled modes influence the reduced order controller and can even cause unstable behaviour [25]. Output filtering with a digital low pass filter reduces these spill-over problems. These filters remove higher order oscillations from the measured output signals. They extend the continuous time physical system and the model with their discrete transfer function.

3.4 State feedback control

In section 3.2 a physical model of a one–link flexible robot moving in the horizontal plane is derived. This model contains physically meaningful parameters as inertia, eigenpulsation, damping, ... Section 3.3 describes the identification of standard discrete linear state space models. For linear state feedback control it is not necessary to transform the identified model to its physical equivalent. The knowledge of the parameters is not needed. (In contrast to linear state feedback control, for nonlinear control design, described in chapter 4, the parameters in the model have to be identified.)

The state feedback control design described in this section is based on the discrete time model:

$$\begin{aligned} \mathbf{x}(k+1) = &\quad \mathbf{F}\mathbf{x}(k) + \mathbf{G}T(k) \\ \mathbf{y}(k) = &\quad \mathbf{H}\mathbf{x}(k) \end{aligned} \tag{3.18}$$

Subsection 3.4.1 describes the design of the state estimator. Subsection 3.4.2 describes the design of the state feedback controller.

3.4.1 The state estimator

Usually the system states cannot be measured directly, and a state estimator is required. There are two basic estimator types: the prediction and the current state estimator.
The prediction estimator generates predictions of the states at instant $k+1$, based only on the information available at instant k (e.g. measurement y(k)).
The current estimator first generates an initial prediction of the states at instant $k+1$, $\tilde{\mathbf{x}}(k+1)$ based only on the information available at instant k. However, at instant $k+1$, after measurement of $\mathbf{y}(k+1)$ it updates these initial estimates to final estimates $\hat{\mathbf{x}}(k+1)$.
At instant k:

$$\tilde{\mathbf{x}}(k+1) = \mathbf{F}\hat{\mathbf{x}}(k) + \mathbf{G}T(k) \tag{3.19}$$

$$\tilde{\mathbf{y}}(k+1) = \mathbf{H}\tilde{\mathbf{x}}(k+1) \tag{3.20}$$

At instant k+1:

$$\hat{\mathbf{x}}(k+1) = \tilde{\mathbf{x}}(k+1) + \mathbf{L}\left[\mathbf{y}(k+1) - \tilde{\mathbf{y}}(k+1)\right] \tag{3.21}$$

The current estimator estimates the state variables more accurately and it reacts faster to external disturbances than a prediction estimator. Therefore in the examples current estimators are used.
The feedback gains of the state estimator are calculated using a pole placement algorithm. The choice of the estimator poles is not so critical. A rule of thumb is that the estimator bandwidth must be higher than the closed loop bandwidth of the controlled system, so that the dynamics of the state estimator have no dominant influence on the controlled system. The estimator poles are tuned experimentally so that a good correspondence between measured and estimated output is obtained.

3.4.2 The control design

The output of a state feedback controller during a tracking operation consists of an acceleration feedforward and a feedback of the difference between the measured and the desired state variables:

$$T(k) = T_{ff}(k) - \sum_{i=1}^{n} K_i(x_i(k) - x_{id}(k)) \tag{3.22}$$

where n is the order of the system. The control design consists of the calculation of: 1) the feedback gains K_i, 2) the feedforward term $T_{ff}(k)$, and 3) the desired state variables $x_{id}(k)$.

3.4.2.1 Calculation of feedback gains

Several criteria can be handled to calculate the state feedback gains [7],[27]. Pole placement seems to be the most straightforward approach for flexible robots [63]. It allows a fine tuning of the closed loop dynamics. In order to choose the closed loop poles the following considerations have to be made:

- Only the poles of the open loop system that result in a bad dynamic behaviour under tracking conditions have to be replaced. This strategy avoids unnecessary large feedback gains. For example, system poles that originate from the filter dynamics can be copied to the closed loop system.
- For good tracking dynamics the dominant poles are chosen as

$$\omega_0 = 2\pi f \quad \xi = 1$$

and for good regulator dynamics

$$\omega_0 = 2\pi f \quad \xi = 0.707$$

f is the frequency of the poles [23].

In optimal control, the feedback gain matrix K is determined such that a quadratic performance index,

$$J = \sum_{k=0}^{\infty} (\mathbf{x}^T(k)\mathbf{Q}\mathbf{x}(k) + T^T(k)\mathbf{R}T(k))$$

is minimized. The weighting matrices Q and R are positive–definite real symmetrix matrices which determine the weight given on the different state variables and on the control input. However, the relation between the selected Q and R and the obtained closed loop dynamics remains very obscure, such that in practice it is not at all straightforward to come up with a suitable Q and R.

3.4.2.2 Feedforward calculation

The feedforward term is determined as being the input needed to obtain exact correspondence between output, y = y_e, and desired trajectory, y_{ed}. This involves calculation of the inverse model 3.4. The transfer function

$$\frac{y_e}{T}(s) = \frac{l}{I_t s^2} + \frac{1}{I_t} \sum_{i=1}^{n} \frac{\Psi_i(l)\frac{d\Psi_i}{dx''}(0)}{s^2 + 2\zeta_i\omega_i s + \omega_i^2} \tag{3.23}$$

(calculated from 3.4 and 3.6) may contain unstable zero's (nonminimum phase). This is a typical property of noncollocated systems where a mechanical system is controlled by applying control torques at one end that are based on sensor signals measured at the other end. Cannon and Schmitz have described this property for one–flexible link structures in [11]. The unstable zeroes become unstable poles in the inverse model, and cause stability problems in the inverse model calculation.

Because the calculation of the inverse model 3.23 gives stability problems only acceleration feedforward is introduced. The acceleration feedforward is given by

$$T_{ff} = N_a \ddot{y}_{ed} \tag{3.24}$$

The constant N_a is determined as the low frequency gain on the Bode diagram of transfer function 3.23. This means that only rigid body dynamics are considered; structural dynamics are neglected.
The constant N_a is determined as

$$\frac{1}{N_a} = \frac{\ddot{y}_{ed}}{T_{ff}} = \lim_{s \to 0} s^2 \frac{y_e}{T}(s) = \frac{l}{I_l} \tag{3.25}$$

or from the discrete model 3.18 as

$$\frac{1}{N_a} = \frac{\ddot{y}_{ed}}{T_{ff}} = \lim_{z \to 1} (1 - z^{-1})^2 \frac{y_e}{T}(z) = \lim_{z \to 1} (1 - z^{-1})^2 \, \mathbf{H}(z\mathbf{I} - \mathbf{F})^{-1}\mathbf{G} \tag{3.26}$$

Further investigations about feedforward calculation by Torfs [72] show the possibility of quasi–perfect feedforward calculation of non–minimal phase systems. An algorithm is proposed, which is based on the zero phase error tracking control algorithm of Tomizuka [71]. The algorithm cancels all system poles and zeros, without introducing any phase error. A prefilter based on the inverse system model can then be used.

3.4.2.3 Desired state calculation

The desired state variables x_{id} can be calculated by exciting the identified model with input T_{ff}. Clearly, this input will excite the resonance frequency which results in an undesirable trajectory for the states. Therefore in the calculation of desired states only rigid body dynamics and deflections due to inertial rigid body forces (quasi–static deflections) are considered. An example of desired state calculation is given in chapter 6 "Experimental results on a flexible one–link robot".

3.5 Limitations of linear control techniques for flexible robots with nonlinear joint friction

Nonlinear joint friction has a large effect on the design of a linear controller for a flexible robot. Nonlinear friction causes limit cycles and steady state errors, which cannot be corrected with plain state feedback. Limit cycles inhibit accurate positioning and appear when the feedback gains are too high. These limit cycles are avoided by

restricting the closed loop bandwidth. Also a strategy is proposed to reduce the limit cycles by phasing out the feedback once the final position is reached. Steady state errors are eliminated with integral feedback.

3.5.1 Reduction of the steady state error by integral action

Steady state errors are caused by nonlinear motor friction which makes the motor stop before the desired position is reached. These steady state errors are not corrected using the control law, explained in section 3.4.2.

To eliminate this steady state error an integral term may be added to the controller output:

$$T_i(k) = k_i T_s e(k) + T_i(k-1) \tag{3.27}$$

where k_i is the integral feedback gain
T_s is the sampling period
e(k) is the tracking error

The integral feedback gain, k_i, is determined experimentally.

An alternative approach consists of adding an additional state variable to the state space model. This state variable corresponds to the integral of the position. The integral feedback gain is then calculated together with the state feedback gains using the pole placement algorithm. Comparable results are obtained by using both approaches.

In order to reduce the overshoot caused by the integral action, the integral action is activated only when the position is in the vicinity of the desired final position.

3.5.2 Reduction of limit cycles

Nonlinear friction causes limit cycles if the feedback gains are too high. This effect occurs when the motor comes to a stop in the vicinity of the final position due to the fact that the control torque becomes smaller than the friction torque. The residual energy in the system is then transferred to a limit cycle at a frequency different from the resonance frequencies. The small oscillations are still amplified by adding the integral action to the controller. Accurate positioning becomes impossible then. Therefore on the one hand, the closed loop bandwidth must

be restricted. On the other hand these oscillations can be eliminated by a local modification of the control in the vicinity of the desired final position. The modification consists of phasing out the energy in the system and reinstating the asymptotic stability properties of the open loop system. This is performed by modifying the magnitude of the control input $T(k)$ according to the ratio $V(\mathbf{x}(k))/\delta_m$ when the state $\mathbf{x}(k)$ is in the vicinity of the desired final position [56].

$$T^*(k) = \begin{cases} T(k) & V(\mathbf{x}) \geq \delta_m \\ T(k)\frac{V(\mathbf{x})}{\delta_m} & V(\mathbf{x}) < \delta_m \end{cases} \tag{3.28}$$

$V(\mathbf{x})$ is a continuously differentiable locally positive definite function of the controlled states. δ_m is a small positive constant defining the vicinity of the final state $\mathbf{x} = \mathbf{x}_d$. If by an external disturbance, the function $V(\mathbf{x})$ becomes again higher than δ_m the controller is again activated.
The intention to phase out the energy in the system suggests to use the total energy in the controlled modes as function $V(\mathbf{x})$. However, a simpler positive definite function of the controlled states can be used;

$$V(\mathbf{x}) = (\theta - \theta_d)^2 \tag{3.29}$$

This choice gives satisfactory results in the experiments.

As illustration, figure 3.3 shows the effect of adding the phasing out strategy and the integral action to the controller of the two–link test structure with a flexible first joint, described in chapter 8. The tracking errors of the first and second motor position are shown before applying the phasing out strategy and the integral action (top figure), after applying the phasing out strategy (middle figure), and after applying the integral action (bottom figure). After applying the phasing out strategy, the amplitude of the limit cycles observed when the tip position is in the vicinity of the final position is reduced from 1.5 mm to 0.25 mm. After applying the integral action, the steady state error of the final tip position is reduced from 1.5 mm to 0.2 mm.

3.6 Conclusion

The physical model of a flexible one–link robot is derived based on the Lagrangian formulation, as described in chapter 2. The model

is simplified by employing orthogonal functions for the mode shape functions.

Because of the linearity of the model, classical linear identification and control techniques can be used.
Least squares identification of linear models, based on the identification of transfer functions is described.
A state feedback controller with acceleration feedforward is designed. For state feedback control design it is not necessary to transform the identified model to its physical equivalent. The described state feedback controller is based on an arbitrary discrete time model, without reference to the physical model.

Observation spill-over is avoided by filtering the output signals using a second order digital filter. These filters extend the model order.

Acceleration feedforward is introduced in the state feedback controller to reduce tracking errors to negligible values. The calculation of the acceleration feedforward is based on the steady state behaviour of the system.

Nonlinear joint friction causes limit cycles and steady state errors, which can not be corrected with plain state feedback. Positioning errors are eliminated by integral feedback. Limit cycles caused by the nonlinear joint friction are avoided by a restricted closed loop bandwidth. Also a strategy is proposed to reduce limit cycles by phasing out the feedback once the final position is reached.

The linear control techniques described in this chapter are used in the experiments on control of a one link flexible robot (chapter 6).

In chapter 4, the linear strategies, applied to the flexible one–link robot, will be extended to nonlinear techniques for the more complex case.

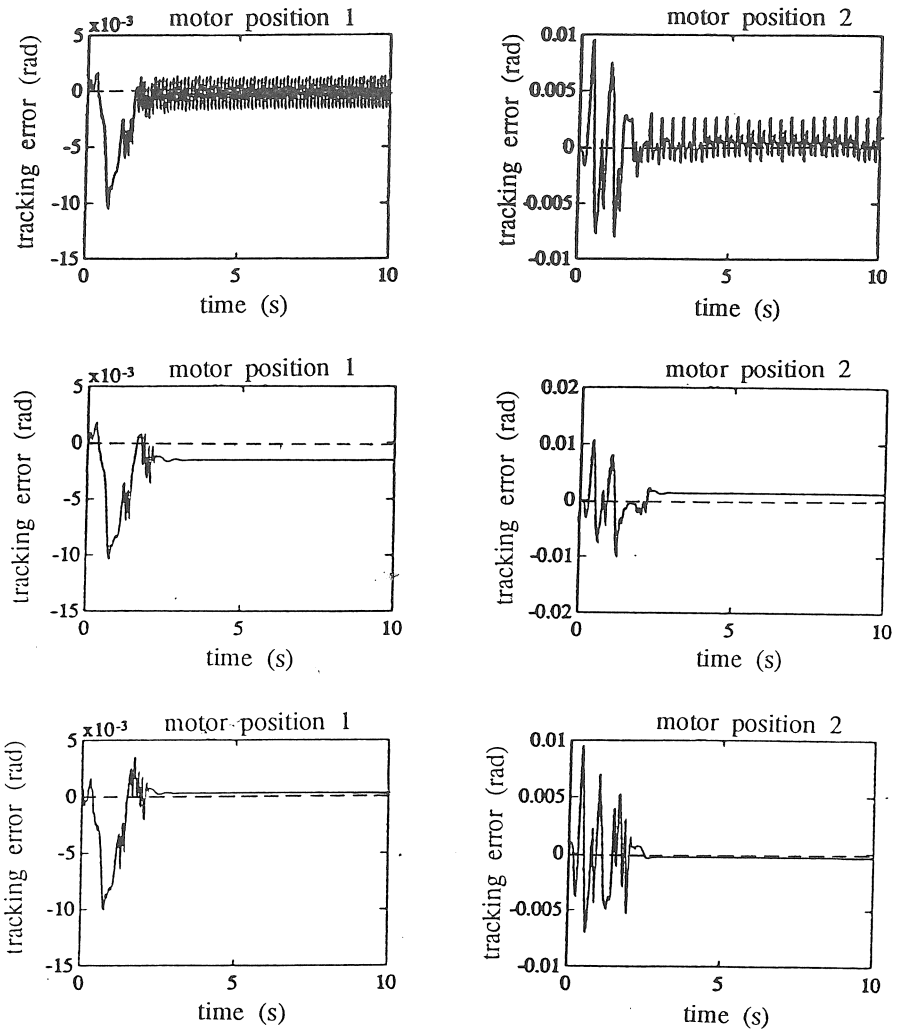

Figure 3.3: Effect of adding the phasing out strategy and the integral action to the controller of the two link test structure: tracking error of the first and second motor position in function of time. Top figure: without application of phasing out strategy and integral action, Middle figure: with application of phasing out strategy, Bottom figure: with application of phasing out strategy and integral action.

Chapter 4

Nonlinear control of multi–link flexible robots

4.1 Introduction

While for one–link flexible manipulators, moving in the horizontal plane, linear control techniques can be used, for multi–link flexible manipulators, the more complex problem of highly nonlinear dynamics and nonlinear couplings between the variables of motion, has to be dealt with.

In this chapter it is first shown that application of the general nonlinear decoupling theory, developed by Freund, to multi–link flexible manipulators does not bring a solution (section 4.4). Then, in section 4.5, the generalized nonlinear decoupling control method is proposed.

The generalized nonlinear decoupling method consists of three stages. The first stage applies a nonlinear decoupling torque to cancel out nonlinear terms in the rigid body dynamics. The greater part of the control torque for trajectory tracking is provided by this dynamics compensation torque. The compensation torque can be a nonlinear feedback torque, calculated based on the measured state variables, or a nonlinear feedforward torque, calculated based on the desired trajectories of the state variables. Section 4.5.3 compares both methods. The second stage consists of linearization of the remaining nonlinearities, and linear feedback of all state variables (rigid and flexible coordinates and their derivatives). This ensures proper stability of the system in the neighbourhood of the desired state. The linear feedback

is assumed to correct for small deviations in the trajectory tracking, and to stabilize excited vibrations. Section 4.5.4 describes the linearization of the model, needed for the computation of the feedback gains.
In a third stage static and dynamic deflection is compensated by taking the flexibility of the structure into account in the desired state computation. This improves the tracking of a reference trajectory. Chapter 5 will describe how the desired state variables can be calculated based on the dynamic model of the flexible robot.

4.2 Notations

The nonlinear equations of motion of a flexible robot are derived by applying Lagrange's equations, as described in chapter 2. They are very complex and can be summarized by:

$$\begin{bmatrix} \mathbf{M}_{11}(\boldsymbol{\theta}_l) & \mathbf{M}_{12}(\boldsymbol{\theta}_l) \\ \mathbf{M}_{12}^T(\boldsymbol{\theta}_l) & \mathbf{M}_{22}(\boldsymbol{\theta}_l) \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}}_0 \\ \ddot{\mathbf{q}}_f \end{bmatrix} + \begin{bmatrix} \mathbf{F}_{\mathbf{q}_0}(\mathbf{x}) \\ \mathbf{F}_{\mathbf{q}_f}(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} \mathbf{T} \\ 0 \end{bmatrix} \tag{4.1}$$

where
$\mathbf{q}_0 = [q_{01} \ \ldots \ q_{0N}]^T \in \Re^N$ is the vector of rigid body coordinates (the motor positions).
$\mathbf{q}_f = [q_{f1} \ \ldots \ q_{fP}]^T \in \Re^P$ is the vector of generalized coordinates associated with joint and/or link flexibility.
Only the dependence of the mass matrix

$$\mathbf{M}(\boldsymbol{\theta}_l) = \begin{bmatrix} \mathbf{M}_{11}(\boldsymbol{\theta}_l) & \mathbf{M}_{12}(\boldsymbol{\theta}_l) \\ \mathbf{M}_{12}^T(\boldsymbol{\theta}_l) & \mathbf{M}_{22}(\boldsymbol{\theta}_l) \end{bmatrix}$$

on the vector of link positions $\boldsymbol{\theta}_l = [\theta_{l1} \ \ldots \ \theta_{lN}]^T \in \Re^N$ is considered. Since the mass matrix is symmetric positive definite, its inverse is also symmetric positive definite and is defined as

$$\mathbf{W}(\boldsymbol{\theta}_l) = \mathbf{M}^{-1}(\boldsymbol{\theta}_l) = \begin{bmatrix} \mathbf{W}_{11}(\boldsymbol{\theta}_l) & \mathbf{W}_{12}(\boldsymbol{\theta}_l) \\ \mathbf{W}_{12}^T(\boldsymbol{\theta}_l) & \mathbf{W}_{22}(\boldsymbol{\theta}_l) \end{bmatrix}$$

The state vector is denoted by:

$$\mathbf{x} = [\mathbf{q}_0 \ \ \mathbf{q}_f \ \dot{\mathbf{q}}_0 \ \ \dot{\mathbf{q}}_f]^T$$

Given a desired trajectory to be followed by the flexible manipulator, the corresponding desired state vector is denoted by

$$\mathbf{x}_d = [\mathbf{q}_{0d} \;\; \mathbf{q}_{fd} \; \dot{\mathbf{q}}_{0d} \;\; \dot{\mathbf{q}}_{fd}]^T$$

The error $\Delta\mathbf{x} = \mathbf{x} - \mathbf{x}_d$ is defined by

$$\Delta\mathbf{x} = [\Delta\mathbf{q}_0 \;\; \Delta\mathbf{q}_f \; \Delta\dot{\mathbf{q}}_0 \;\; \Delta\dot{\mathbf{q}}_f]^T$$

4.3 Classical rigid nonlinear control and its limits.

Classical rigid control methods are implemented in the different examples, to show the limits of these rigid control methods and to demonstrate the improvements obtained by appying nonlinear flexible control methods. The control technique used is the nonlinear decoupling technique [28]. This technique consists of a nonlinear feedback which, in the case of a rigid robot, reduces the dynamic model to a set of linear decoupled subsystems. Each of these subsystems is then controlled by a decoupled linear feedback loop.
The control law is obtained using the dynamic model of a robot with rigid links and joints

$$\mathbf{M}_{11}(\mathbf{q}_0)\ddot{\mathbf{q}}_0 + \mathbf{F}_{\mathbf{q}_0}(\mathbf{q}_0, \dot{\mathbf{q}}_0) = \mathbf{T} \tag{4.2}$$

This model can also be obtained from the flexible model 4.1 by considering only the equations describing the rigid part of the model (the equation in $\mathbf{q}_0$), and by setting the flexible state coordinates $(\mathbf{q}_f, \dot{\mathbf{q}}_f)$ equal to zero. The function $\mathbf{F}_{\mathbf{q}_0}$ is now only function of $\mathbf{q}_0$ and $\dot{\mathbf{q}}_0$.
The control torque consists of two parts:

$$\mathbf{T} = \mathbf{T}_1 + \mathbf{T}_2 \tag{4.3}$$

The first part is the nonlinear feedback:

$$\mathbf{T}_1 = \mathbf{F}^*_{\mathbf{q}_0}(\mathbf{q}_0, \dot{\mathbf{q}}_0) \tag{4.4}$$

with $\mathbf{F}^*_{\mathbf{q}_0}(\mathbf{q}_0, \dot{\mathbf{q}}_0)$ an estimate of $\mathbf{F}_{\mathbf{q}_0}(\mathbf{q}_0, \dot{\mathbf{q}}_0)$. Together with the introduction of an estimate of the matrix $\mathbf{M}_{11}(\mathbf{q}_0)$, $\mathbf{M}^*_{11}(\mathbf{q}_0)$, in the

control loop, the system becomes linear and decoupled.
Indeed, when

$$\mathbf{T}_2 = \mathbf{M}_{11}^*(\mathbf{q}_0)(\ddot{\mathbf{q}}_{0d} + \mathbf{K}_v(\dot{\mathbf{q}}_{0d} - \dot{\mathbf{q}}_0) + \mathbf{K}_p(\mathbf{q}_{0d} - \mathbf{q}_0)) \tag{4.5}$$

and when $\mathbf{M}_{11}^*(\mathbf{q}_0)$ and $\mathbf{F}_{\mathbf{q}_0}^*(\mathbf{q}_0, \dot{\mathbf{q}}_0)$ are perfect estimates of $\mathbf{M}_{11}(\mathbf{q}_0)$ and $\mathbf{F}_{\mathbf{q}_0}(\mathbf{q}_0, \dot{\mathbf{q}}_0)$, the closed loop dynamic equation, obtained by combining 4.2, 4.3, 4.4, and 4.5 becomes:

$$(\ddot{\mathbf{q}}_{0d} - \ddot{\mathbf{q}}_0) + \mathbf{K}_v(\dot{\mathbf{q}}_{0d} - \dot{\mathbf{q}}_0) + \mathbf{K}_p(\mathbf{q}_{0d} - \mathbf{q}_0) = 0$$

where
$\mathbf{q}_{0d} = [q_{01d}\ q_{02d}\ \dots\ q_{0Nd}]^T$ is the vector of the desired joint positions,
$\dot{\mathbf{q}}_{0d} = [\dot{q}_{01d}\ \dot{q}_{02d}\ \dots\ \dot{q}_{0Nd}]^T$ is the vector of the desired joint velocities,
$\ddot{\mathbf{q}}_{0d} = [\ddot{q}_{01d}\ \ddot{q}_{02d}\ \dots\ \ddot{q}_{0Nd}]^T$ is the vector of the desired joint accelerations,
$\mathbf{K}_p$ and $\mathbf{K}_v$ are diagonal matrices. They consists of the position- and velocity feedback gains. By chosing $\mathbf{K}_p$ and $\mathbf{K}_v$ appropriately, the desired dynamic behaviour for the errors $\mathbf{q}_{0d} - \mathbf{q}_0$ can be obtained.
The control scheme is given in figure 4.1.

When the bandwidth of the controller is sufficiently lower than the

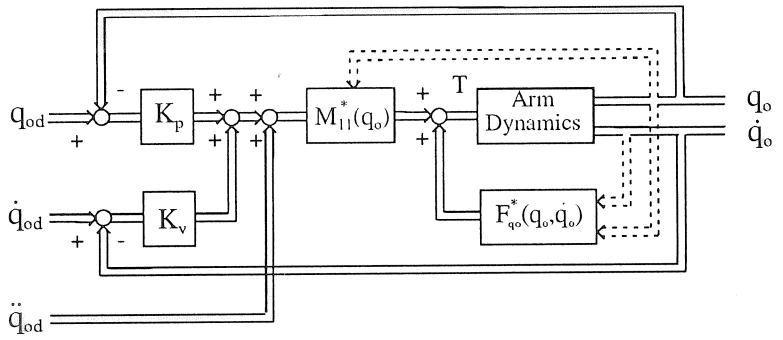

Figure 4.1: Block diagram of the rigid nonlinear controller.

first resonance frequency, this control method gives good results. However, excited oscillations are only weakly damped and static and dynamic deflection causes steady state and tracking errors. When the bandwidth of the controller reaches the system bandwidth the closed loop dynamic behaviour detoriates very rapidly. The system's behaviour will be very oscillatory.

4.4 The nonlinear decoupling theory.

The classical rigid nonlinear control method described in section 4.3 is a special case of the more general nonlinear decoupling theory developed by Freund [28]. The general nonlinear decoupling theory is summarized in appendix A. The aim of nonlinear decoupling is to find the torque to be applied at each motor which makes the robot appear as a linear multivariable system for which a simple linear controller is then designed. In order to explain this method the concept of differential order needs to be defined. In appendix A an exact mathematical definition is given. A more intuitive explanation is that the differential order is the highest order of the derivative of the output variable in the system equations. Applying the general nonlinear control theory to 1) a rigid robot, 2) a flexible link robot and 3) a flexible joint robot leads to the following results:

1. For a rigid robot the differential order of the output (the motor position $\mathbf{q}_0$) is two. The order of the system is 2N, with N the number of degrees of freedom of the robot. For each input-variable, a linear subsystem of order two (the differential order) can be separated, by using the nonlinear decoupling technique. This results in N decoupled subsystems. Each subsystem can be controlled independently by one input variable. This approach is used in section 4.3.

2. For a flexible link robot the differential order of the outputs is also equal to two. (The output can be the motor position $\mathbf{q}_0$, a flexible coordinate, or the tip position). The order of the flexible dynamic system is 2N+2P, with P the number of generalized coordinates associated with the link flexibility. There are again N input variables. Hence, by using the nonlinear decoupling technique, N decoupled subsystems of order 2, can be separated and controlled. The remaining part of the system (order 2P) cannot be controlled.
   As an example, the nonlinear decoupling control applied to a one–link flexible robot arm is simulated. The numerical values of the parameters of the simulation model (equation 3.7) are summarized in table 4.1. The bandwidth of the nonlinear decoupling controller is 50 Hz. Appendix A, derives the nonlinear

decoupling control law for the one–link flexible robot arm. There is only one input variable (the torque), hence only one subsystem can be controlled. The remaining part cannot be controlled. This is shown mathematically in appendix A. If one is only interested in the tip position of the robot, this variable can be chosen as output. The tip can then be controlled to follow the desired trajectory exactly. On the other hand the computed torque profiles will be oscillatory. This will cause undesired vibrations when these torques are used to control the arm. The simulation results are given in figure 4.2. The left figures show the simulated and desired displacement of the tip, the motor torque and the motor position. The right figures are a detail of the left figures. The tip position has the chosen dynamics,

$$\ddot{y}_e(t) + \alpha_1 \dot{y}_e(t) + \alpha_0 y_e(t) = \lambda w(t)$$

with $y_e(t)$ the tip position,
w(t) the desired tip position,
$\alpha_0 = (2\ \pi\ 50)^2\ \ 1/s^2$,
$\alpha_1 = 2\ (2\ \pi\ 50)\ \ 1/s$,
$\lambda = \alpha_0$.

The torque and the motor position are oscillatory (middle and bottom figures).

3. For a flexible joint robot the differential order of the outputs (the link positions or joint angles after the elastic transmissions) is four. The order of the system is 4N (N rigid degrees of freedom, N flexible degrees of freedom: each joint is modeled as a linear spring). For each input variable a linear subsystem of order four (the differential order) can be separated, by using the nonlinear decoupling technique. Each subsystem can be controlled independently by one inputvariable. Hence, a torque can be calculated which decouples and linearizes the system dynamics (see appendix B). This solution, obtained by applying the general nonlinear decoupling theory, is the same solution as found by Bortoff and Spong [65].

As a conclusion nonlinear decoupling theory can be applied to rigid robots and to flexible joint robots. Application of nonlinear decoupling

to flexible link robots does not bring a solution. The obtained control method for flexible joint robots has the disadvantage that in addition to link positions and velocities also link accelerations and jerks are needed for feedback. Because measurements of link acceleration and jerk are usually not available, there is need for a nonlinear state estimator. Another possibility is to compute the accelerations and jerks by differentiation. Because of the sensitivity to high frequency noise of the differentiation algorithms, especially for the calculation of higher derivatives, this is not suited in practice (see chapter 5).

length l:	$0.5\ m$
cross section S:	$0.00032\ m^2$
geometric moment	
of inertia I:	$0.17110^{-8}\ m^4$
Young modulus E:	$0.7\ 10^{11}\ N/m^2$
density ρ:	$2800\ kg/m^3$
mass m:	$0.448\ kg$
inertia I_b:	$0.0373\ kgm^2$
inertia of the hub I_h:	$0.79\ kgm^2$
total inertia I_t :	$0.8273\ kgm^2$
no payload mass	
the motor friction is neglected	
one flexible mode is considered:	
resonance frequency ω:	$162.7\ rad/sec$
damping ratio ζ:	0.03
$\psi(l)$:	$1.9245\ m$
$\frac{d\psi}{dx^n}(0)$:	$-0.1542\ rad$

Table 4.1: numerical parameters of the simulation model of a one link flexible robot arm

4.5 The generalized nonlinear decoupling theory.

The use of standard nonlinear control techniques (section 4.3) is limited as soon as the control law bandwidth approaches the lowest structural resonance frequency. Application of nonlinear decoupling to the complete system dynamics of flexible link robots (rigid body + flexible

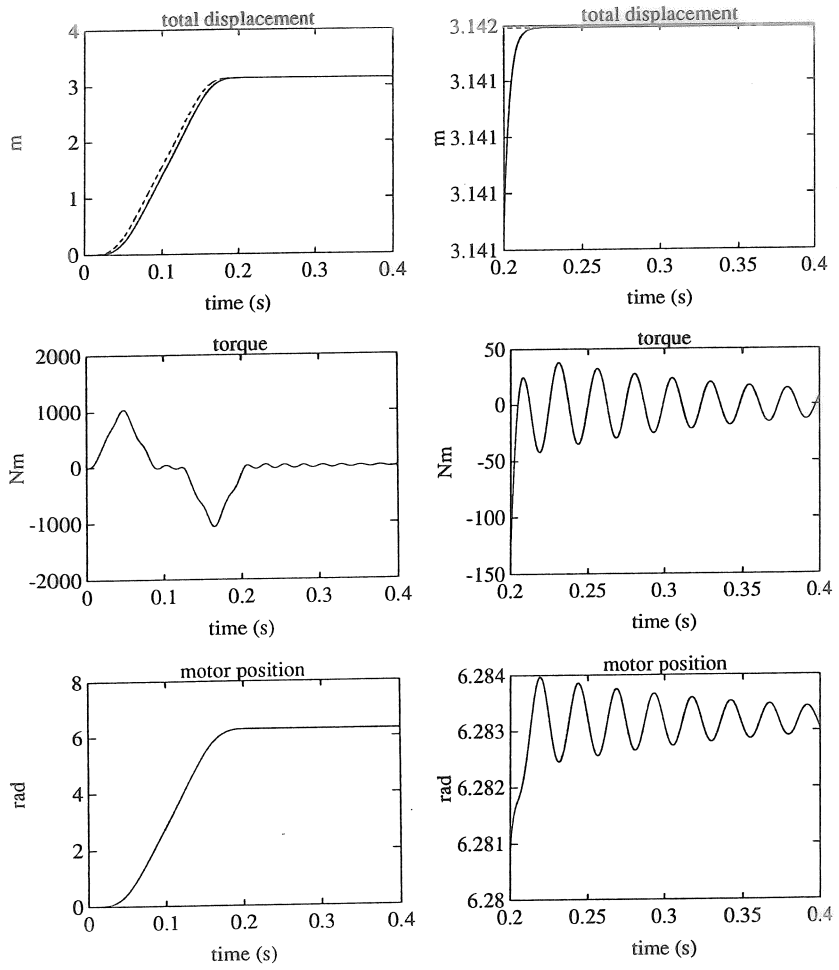

Figure 4.2: Control of a one link flexible robot arm using the nonlinear decoupling technique (simulation results). Upper figure: simulated (solid line) and desired (dashed line) displacement of the tip. Middle figure: torque. Bottom figure: motor position. The right figures are a detail of the left figures.

dynamics) does not bring a solution, since there are insufficient independent control inputs in order to compensate all nonlinearities in the system (there is only one actuator per joint, but there is an infinite number of degrees of freedom) (section 4.4). The independent control inputs may be used to decouple and linearize the rigid body dynamics, but the flexible dynamics will then become uncontrollable.
For flexible joint robots a solution can be found using the standard nonlinear decoupling theory. However the obtained control method has the disadvantage that in addition to link positions and velocities also link accelerations and jerks are needed for feedback.

The generalized nonlinear decoupling method has been suggested by Froment and Faillot [20], [29] and is similar to the method proposed by Singh and Schy [64]. It uses nonlinear decoupling to obtain feedback linearization of the state variables containing the rigid part (in an inner loop), but enables to control the full state of the system (in an outer loop). The method is applicable to flexible link robots as well as to flexible joint robots. For flexible joint robots, the method differs from the standard nonlinear decoupling method in the choice of the feedback variables. Link accelerations and jerks are not needed, using the generalized nonlinear decoupling method.

Subsection 4.5.1 gives the generalized nonlinear control law. The control law is theoretically justified by applying the nonlinear decoupling formalism in subsection 4.5.2 . Subsection 4.5.3 explains why in the experiments the nonlinear decoupling torque is replaced by a nonlinear feedforward torque. Subsection 4.5.4 explains the computation of feedback gains, needed for linear feedback of state variables in the outer loop. Appendix B makes a comparison of the generalized nonlinear control law and the nonlinear control law obtained by applying the nonlinear decoupling theory to flexible joint robots.

4.5.1 Control law design

The control law is split up into two feedback loops.

First loop: the inner loop (non linear) which realizes feedback linearization of the state variables containing the rigid part:

$$\mathbf{T} = \mathbf{F}^*_{\mathbf{q}_0}(\mathbf{x}) + \mathbf{W}_{11}^{-1*}\mathbf{W}^*_{12}(\theta_l)\,\mathbf{F}^*_{\mathbf{q}_f}(\mathbf{x}) + \mathbf{W}_{11}^{-1*}(\theta_l)\mathbf{T}_1 \qquad (4.6)$$

with $\mathbf{F}^*_{\mathbf{q}_0}$, $\mathbf{F}^*_{\mathbf{q}_f}$, $\mathbf{W}^*_{11}$ and $\mathbf{W}^*_{12}$ estimates of $\mathbf{F}_{\mathbf{q}_0}$, $\mathbf{F}_{\mathbf{q}_f}$, $\mathbf{W}_{11}$ and $\mathbf{W}_{12}$.

In the sequel it is supposed that the vector functions $\mathbf{F}^*$ and the matrices $\mathbf{W}^*$ are perfect estimates of $\mathbf{F}$ and $\mathbf{W}$. Hence, for ease of notation, the superscript $*$ is neglected.

Then, equation 4.1 becomes:

$$\begin{aligned} \ddot{\mathbf{q}}_0 = &\quad \mathbf{T}_1 && (linear) \\ \ddot{\mathbf{q}}_f = &\quad (\mathbf{W}_{12}^T\mathbf{W}_{11}^{-1}\mathbf{W}_{12} - \mathbf{W}_{22})(\theta_l)\,\mathbf{F}_{\mathbf{q}_f}(\mathbf{x}) + \mathbf{W}_{12}^T\mathbf{W}_{11}^{-1} && (\theta_l)\mathbf{T}_1 \quad (4.7) \\ & && (nonlinear) \end{aligned}$$

Second loop: the outer loop (linear), which ensures proper stability of the system about desired state $\mathbf{x}_d$, by linear feedback of the full state $\mathbf{x}$:

$$\mathbf{T}_1 = \ddot{\mathbf{q}}_{0d} - \mathbf{K}(\mathbf{x} - \mathbf{x}_d) \tag{4.8}$$

where $\mathbf{K} = [\mathbf{K}_{\mathbf{q}_0}\ \mathbf{K}_{\mathbf{q}_f}\ \mathbf{K}_{\dot{\mathbf{q}}_0}\ \mathbf{K}_{\dot{\mathbf{q}}_f}]$ is a N x (2N + 2P) feedback gain matrix.

Then, after combining 4.8 and 4.8, the closed loop equations of the system become

$$\begin{aligned} \Delta\ddot{\mathbf{q}}_0 + &\quad \mathbf{K}\Delta\mathbf{x} = 0 && (linear) \\ \Delta\ddot{\mathbf{q}}_f + &\quad \mathbf{W}_{12}^T\mathbf{W}_{11}^{-1}(\theta_l)\,\mathbf{K}\,\Delta\mathbf{x} = && \quad (4.9) \\ & \quad (\mathbf{W}_{12}^T\mathbf{W}_{11}^{-1}\mathbf{W}_{12} - \mathbf{W}_{22})(\theta_l)\,\mathbf{F}_{\mathbf{q}_f}(\mathbf{x}) \\ & \quad -\ddot{\mathbf{q}}_{fd} + \mathbf{W}_{12}^T\mathbf{W}_{11}^{-1}(\theta_l)\,\ddot{\mathbf{q}}_{0d} && (nonlinear) \end{aligned}$$

The complete feedback control law is obtained from equations 4.6 and 4.8 as

$$\mathbf{T} = \mathbf{F}_{\mathbf{q}_0}(\mathbf{x}) + \mathbf{W}_{11}^{-1}\mathbf{W}_{12}(\theta_l)\,\mathbf{F}_{\mathbf{q}_f}(\mathbf{x}) + \mathbf{W}_{11}^{-1}(\theta_l)\ddot{\mathbf{q}}_{0d} - \mathbf{W}_{11}^{-1}(\theta_l)\mathbf{K}(\mathbf{x} - \mathbf{x}_d) \tag{4.10}$$

Figure 4.3 gives the corresponding block diagram.

4.5.2 Theoretical justification of the control law design.

The control law 4.10 is derived by applying the non linear decoupling formalism (see appendix A). This is demonstrated in this section. The dynamic equations 4.1 of a flexible manipulator can be written in state space form as:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}(\mathbf{x}, t) + \mathbf{B}(\mathbf{x}, t)\,\mathbf{T} \\ \mathbf{y} &= \mathbf{C}(\mathbf{x}, t) \end{aligned} \tag{4.11}$$

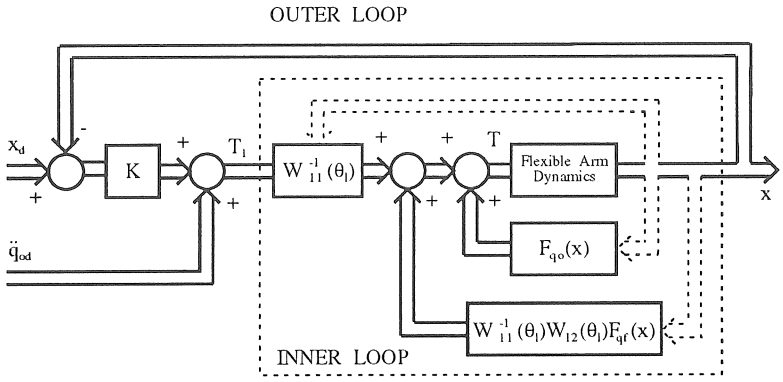

Figure 4.3: Block diagram of the nonlinear controller.

with

$$\mathbf{x} = [\, \mathbf{q}_0 \;\; \mathbf{q}_f \;\; \dot{\mathbf{q}}_0 \;\; \dot{\mathbf{q}}_f \,]^T$$

$$\mathbf{A}(\mathbf{x},t) = \begin{bmatrix} \dot{\mathbf{q}}_0 \\ \dot{\mathbf{q}}_f \\ -\mathbf{W}_{11}(\theta_l)\mathbf{F}_{\mathbf{q}_0}(\mathbf{x}) - \mathbf{W}_{12}(\theta_l)\mathbf{F}_{\mathbf{q}_f}(\mathbf{x}) \\ -\mathbf{W}_{12}^T(\theta_l)\mathbf{F}_{\mathbf{q}_0}(\mathbf{x}) - \mathbf{W}_{22}(\theta_l)\mathbf{F}_{\mathbf{q}_f}(\mathbf{x}) \end{bmatrix}$$

$$\mathbf{B}(\mathbf{x},t) = \begin{bmatrix} 0 \\ 0 \\ \mathbf{W}_{11}(\mathbf{q}_0) \\ \mathbf{W}_{12}^T(\mathbf{q}_0) \end{bmatrix}$$

Since decoupling of rigid variables is chosen,

$$\mathbf{C}(\mathbf{x},t) = \mathbf{q}_0$$

The nonlinear decoupling feedback law is given by equations (A.4) of the appendix.

$$\mathbf{T}(\mathbf{x},t) = -\mathbf{D}^{*-1}(\mathbf{x},t)\,\mathbf{C}^*(\mathbf{x},t) + \mathbf{D}^{*-1}(\mathbf{x},t)\,(\Lambda \mathbf{w}(t) - \mathbf{M}^*(\mathbf{x},t))$$

where $\mathbf{w}(t)$ is the reference trajectory for the output variable. A definition of the matrices $\mathbf{D}^*$, $\mathbf{C}^*$ and $\mathbf{M}^*$ is given in the appendix. The control law is made of two terms:

- a torque $-\mathbf{D}^{*-1}(\mathbf{x},t)\ \mathbf{C}^*(\mathbf{x},t)$ completely defined by the open loop behaviour of the system. In this case it is equal to

$$\mathbf{F}_{\mathbf{q}_0}(\mathbf{x}) + \mathbf{W}_{11}^{-1}\mathbf{W}_{12}(\theta_l)\ \mathbf{F}_{\mathbf{q}_f}(\mathbf{x})$$

- a torque $\mathbf{D}^{*-1}(\mathbf{x},t)\ (\Lambda\mathbf{w}(t) - \mathbf{M}^*(\mathbf{x},t)) = \mathbf{W}_{11}^{-1}(\theta_l)\ (\Lambda w(t) - \mathbf{M}^*(\mathbf{x},t))$ which mainly depends on the dynamics chosen for the closed loop system, through Λ and $\mathbf{M}^*(\mathbf{x},t)$.

Instead of choosing Λ, $\mathbf{w}$ and $\mathbf{M}^*$ in order to achieve decoupling of the transfer function relating the rigid variable to its reference value, it is selected to ensure convenient closed loop dynamics to the whole state by choosing

$$\begin{aligned}\Lambda &= \mathbf{I}\\ \mathbf{w}(t) &= \ddot{\mathbf{q}}_{0d}(t)\\ \mathbf{M}^*(\mathbf{x},t) &= \mathbf{K}(\mathbf{x}-\mathbf{x}_d)\end{aligned}$$

which precisely lead to equation 4.10 and enable simultaneous

- linearization of rigid variables equations
- control of the whole state (rigid and flexible parts) by a proper choice of the gain matrix $\mathbf{K}$.

4.5.3 Nonlinear feedback versus nonlinear feedforward.

A nonlinear feedforward scheme compensates for the manipulator dynamics in the feedforward path while a nonlinear feedback scheme uses the dynamics in the feedback loop for linearization and decoupling.

For rigid manipulators, Khosla and Kanada ([44]) have experimentally evaluated nonlinear feedback and feedforward control schemes. They concluded that if an exact model of the maniplator is available then both the feedback and feedforward compensation scheme will give similar results. In this case, using the feedforward dynamics compensation may have some implementational advantages because the feedforward torques can be computed off–line and added on–line to the torques computed by the feedback controllers.

This conclusion can be extended to the generalized nonlinear decoupling control of flexible robots. Another advantage observed in the

experiments is that using the noise free desired values results in a more stable behaviour of the controller. Because of the advantages of applying the feedforward scheme, in the experiments (chapter 8 and 9) nonlinear feedforward is used. The control law then becomes:

$$\begin{aligned}\mathbf{T} = \quad & \mathbf{F}_{\mathbf{q}_0}(\mathbf{x}_d) + \mathbf{W}_{11}^{-1}\mathbf{W}_{12}(\theta_{ld})\,\mathbf{F}_{\mathbf{q}_f}(\mathbf{x}_d) + \mathbf{W}_{11}^{-1}(\theta_{ld})\ddot{\mathbf{q}}_{0d} \\ & -\mathbf{W}_{11}^{-1}(\theta_{ld})\mathbf{K}(\mathbf{x}-\mathbf{x}_d)\end{aligned} \tag{4.12}$$

or

$$\mathbf{T} = \mathbf{T}_{ff} + \mathbf{W}_{11}^{-1}(\theta_{ld})\mathbf{T}_{fb} \tag{4.13}$$

with

$$\mathbf{T}_{ff} = \quad \mathbf{F}_{\mathbf{q}_0}(\mathbf{x}_d) + \mathbf{W}_{11}^{-1}\mathbf{W}_{12}(\theta_{ld})\,\mathbf{F}_{qf}(\mathbf{x}_d) + \mathbf{W}_{11}^{-1}(\theta_{ld})\ddot{\mathbf{q}}_{0d} \tag{4.14}$$

$$\mathbf{T}_{fb} = \quad -\mathbf{K}\,(\mathbf{x}-\mathbf{x}_d) \tag{4.15}$$

4.5.4 Feedback gains computation

The second stage of the generalized nonlinear decoupling controller is a linear feedback of all state variables (rigid and flexible coordinates and their derivatives). In order to use linear techniques (pole placement, optimal control) for feedback gains computation, the model needs to be linearized to the first order about the desired trajectory. Two approaches are possible:

- The first approach is the linearization in a number of points along the trajectory. The feedback gain $\mathbf{K}$ will be function of the position. In this approach a large computing time is needed.
- Since the main purpose in flexible manipulator control is to damp out the deflections in the final position as fast as possible, another approach is to perform the linearization only in the final position. In this way the linearization and calculation of the feedback gains for each position can be avoided (provided that the system will not go unstable along the trajectory).

A strategy which can be followed in practical implementations is to table a number of feedback gains as a function of the configuration of the robot. These configurations (linearization points) have to be determined based on a sensitivity analysis, which gives the range of

points for which the controller with feedback gains calculated in a certain configuration, remains stable. The gain at a given configuration is then found by table look–up.

The linearized model is given by:

$$\begin{bmatrix} \mathbf{M}_{11}(\boldsymbol{\theta}_{ld}) & \mathbf{M}_{12}(\boldsymbol{\theta}_{ld}) \\ \mathbf{M}_{12}^T(\boldsymbol{\theta}_{ld}) & \mathbf{M}_{22}(\boldsymbol{\theta}_{ld}) \end{bmatrix} \begin{bmatrix} \Delta\ddot{\mathbf{q}}_0 \\ \Delta\ddot{\mathbf{q}}_f \end{bmatrix} + \begin{bmatrix} \frac{\delta \mathbf{F}_{\mathbf{q}_0}}{\delta \mathbf{x}}|_{\mathbf{x}_d} \\ \frac{\delta \mathbf{F}_{\mathbf{q}_f}}{\delta \mathbf{x}}|_{\mathbf{x}_d} \end{bmatrix} \Delta\mathbf{x} = \begin{bmatrix} \Delta\mathbf{T} \\ 0 \end{bmatrix} \tag{4.16}$$

with

$$\mathbf{M}_{ij}(\boldsymbol{\theta}_l) = \mathbf{M}_{ij}(\boldsymbol{\theta}_{ld}), \qquad i = 1,2 \quad j = 1,2$$

which implies that the mass matrix varies slowly

$$\begin{aligned} \mathbf{F}_{\mathbf{q}_0}(\mathbf{x}) &= \mathbf{F}_{\mathbf{q}_0}(\mathbf{x}_d) + \frac{\delta \mathbf{F}_{\mathbf{q}_0}}{\delta \mathbf{x}}|_{\mathbf{x}_d} \Delta\mathbf{x}, \\ \mathbf{F}_{\mathbf{q}_f}(\mathbf{x}) &= \mathbf{F}_{\mathbf{q}_f}(\mathbf{x}_d) + \frac{\delta \mathbf{F}_{\mathbf{q}_f}}{\delta \mathbf{x}}|_{\mathbf{x}_d} \Delta\mathbf{x}, \\ \Delta\mathbf{T} &= \mathbf{W}_{11}^{-1} \mathbf{T}_{fb} \end{aligned} \tag{4.17}$$

where $\frac{\delta \mathbf{F}_{\mathbf{q}_0}}{\delta \mathbf{x}}|_{\mathbf{x}_d}$ is a Nx2(N+P) matrix and $\frac{\delta \mathbf{F}_{\mathbf{q}_0}}{\delta \mathbf{x}}|_{\mathbf{x}_d}$ is a Px2(N+P) matrix.

Denote

$$\begin{aligned} \mathbf{G}(\mathbf{x}_d) &= -\begin{bmatrix} \mathbf{W}_{11}(\boldsymbol{\theta}_{ld}) & \mathbf{W}_{12}(\boldsymbol{\theta}_{ld}) \\ \mathbf{W}_{12}^T(\boldsymbol{\theta}_{ld}) & \mathbf{W}_{22}(\boldsymbol{\theta}_{ld}) \end{bmatrix} \begin{bmatrix} \frac{\delta \mathbf{F}_{\mathbf{q}_0}}{\delta \mathbf{x}}|_{\mathbf{x}_d} \\ \frac{\delta \mathbf{F}_{\mathbf{q}_f}}{\delta \mathbf{x}}|_{\mathbf{x}_d} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{G}_{\mathbf{q}_0\mathbf{q}_0} & \mathbf{G}_{\mathbf{q}_0\mathbf{q}_f} & \mathbf{G}_{\mathbf{q}_0\dot{\mathbf{q}}_0} & \mathbf{G}_{\mathbf{q}_0\dot{\mathbf{q}}_f} \\ \mathbf{G}_{\mathbf{q}_f\mathbf{q}_0} & \mathbf{G}_{\mathbf{q}_f\mathbf{q}_f} & \mathbf{G}_{\mathbf{q}_f\dot{\mathbf{q}}_0} & \mathbf{G}_{\mathbf{q}_f\dot{\mathbf{q}}_f} \end{bmatrix} \end{aligned} \tag{4.18}$$

where $\mathbf{G}(\mathbf{x}_d)$ is a (N+P)x2(N+P) matrix.

The linearized model in state space form can now be written from equation 4.16 as

$$\begin{bmatrix} \Delta\dot{\mathbf{q}}_0 \\ \Delta\dot{\mathbf{q}}_f \\ \Delta\ddot{\mathbf{q}}_0 \\ \Delta\ddot{\mathbf{q}}_f \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \\ \mathbf{G}_{\mathbf{q}_0\mathbf{q}_0} & \mathbf{G}_{\mathbf{q}_0\mathbf{q}_f} & \mathbf{G}_{\mathbf{q}_0\dot{\mathbf{q}}_0} & \mathbf{G}_{\mathbf{q}_0\dot{\mathbf{q}}_f} \\ \mathbf{G}_{\mathbf{q}_f\mathbf{q}_0} & \mathbf{G}_{\mathbf{q}_f\mathbf{q}_f} & \mathbf{G}_{\mathbf{q}_f\dot{\mathbf{q}}_0} & \mathbf{G}_{\mathbf{q}_f\dot{\mathbf{q}}_f} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{q}_0 \\ \Delta\mathbf{q}_f \\ \Delta\dot{\mathbf{q}}_0 \\ \Delta\dot{\mathbf{q}}_f \end{bmatrix}$$

$$+ \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{W}_{11}(\theta_{ld}) \\ \mathbf{W}_{12}^T(\theta_{ld}) \end{bmatrix} \Delta\mathbf{T} \tag{4.19}$$

or

$$\Delta\dot{\mathbf{x}} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \\ \mathbf{G}_{\mathbf{q}_0\mathbf{q}_0} & \mathbf{G}_{\mathbf{q}_0\mathbf{q}_f} & \mathbf{G}_{\mathbf{q}_0\dot{\mathbf{q}}_0} & \mathbf{G}_{\mathbf{q}_0\dot{\mathbf{q}}_f} \\ \mathbf{G}_{\mathbf{q}_f\mathbf{q}_0} & \mathbf{G}_{\mathbf{q}_f\mathbf{q}_f} & \mathbf{G}_{\mathbf{q}_f\dot{\mathbf{q}}_0} & \mathbf{G}_{\mathbf{q}_f\dot{\mathbf{q}}_f} \end{bmatrix} \Delta\mathbf{x}$$

$$+ \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{I} \\ \mathbf{W}_{12}^T\mathbf{W}_{11}^{-1}(\theta_{ld}) \end{bmatrix} \mathbf{T}_{fb} \tag{4.20}$$

When nonlinear feedback is used, after having applied the first control loop (feedback linearization of the state variables containing the rigid part) the model 4.2 is yet partially linearized. (Of course, this is only the case if an "exact" model is available.) It is thus not necessary to linearize the full nonlinear model. The feedback gain calculation can be based on the partially linearized model given by equation 4.4 [29].

However, when nonlinear feedforward is used, to provide stability, the feedback gain calculation has to be based on the total linearized model. (Whereas feedback is capable of modifying the system dynamics, this is not the case for feedforward control). In practice, also feedback gain calculation based on the partially linearized model will provide stability if the measured and desired states are approximately equal. This can be obtained if the desired state variables are calculated based on the dynamic model of the flexible robot (see chapter 5).

Denote

$$\mathbf{H}(\mathbf{x}_d) = (\mathbf{W}_{12}^T\mathbf{W}_{11}^{-1}\mathbf{W}_{12} - \mathbf{W}_{22})(\theta_{ld}) \left.\frac{\delta\mathbf{F}_{\mathbf{q}_f}}{\delta\mathbf{x}}\right|_{\mathbf{x}_d}$$

$$\mathbf{H}_o(\mathbf{x}_d) = (\mathbf{W}_{12}^T\mathbf{W}_{11}^{-1}\mathbf{W}_{12} - \mathbf{W}_{22})(\theta_{ld})\, \mathbf{F}_{\mathbf{q}_f}(\mathbf{x}_d) + \mathbf{W}_{12}^T\mathbf{W}_{11}^{-1}(\theta_{ld})\, \ddot{\mathbf{q}}_{0d} - \ddot{\mathbf{q}}_{fd}$$

$$\mathbf{W}_h(\theta_{ld}) = \quad \mathbf{W}_{12}^T \mathbf{W}_{11}^{-1}(\theta_{ld}) \tag{4.21}$$

where

$$\mathbf{H}(\mathbf{x}_d) = \begin{bmatrix} \mathbf{H}_{\mathbf{q}_0}(\mathbf{x}_d) & \mathbf{H}_{\mathbf{q}_f}(\mathbf{x}_d) & \mathbf{H}_{\dot{\mathbf{q}}_0}(\mathbf{x}_d) & \mathbf{H}_{\dot{\mathbf{q}}_f}(\mathbf{x}_d) \end{bmatrix}$$

is a Px2(N+P) matrix and $\mathbf{H}_o(\mathbf{x}_d)$ is a PxP matrix.
Equation 4.4 becomes

$$\begin{aligned} \Delta\ddot{\mathbf{q}}_0 \quad & +\mathbf{K}\Delta\mathbf{x} \quad = \mathbf{0} \\ \Delta\ddot{\mathbf{q}}_f \quad & +\mathbf{W}_h\mathbf{K}\Delta\mathbf{x} \quad = \mathbf{H}_o(\mathbf{x}_d) + \mathbf{H}(\mathbf{x}_d)\Delta\mathbf{x} \end{aligned} \tag{4.22}$$

The linearized closed–loop equations can now be written as

$$\begin{aligned} \Delta\dot{\mathbf{x}} \quad = \quad & \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \\ -\mathbf{K}_{\mathbf{q}_0} & -\mathbf{K}_{\mathbf{q}_f} & -\mathbf{K}_{\dot{\mathbf{q}}_0} & -\mathbf{K}_{\dot{\mathbf{q}}_f} \\ \mathbf{H}_{\mathbf{q}_0}(\mathbf{x}_d) & \mathbf{H}_{\mathbf{q}_f}(\mathbf{x}_d) & \mathbf{H}_{\dot{\mathbf{q}}_0}(\mathbf{x}_d) & \mathbf{H}_{\dot{\mathbf{q}}_f}(\mathbf{x}_d) \\ -\mathbf{W}_h\mathbf{K}_{\mathbf{q}_0} & -\mathbf{W}_h\mathbf{K}_{\mathbf{q}_f} & -\mathbf{W}_h\mathbf{K}_{\dot{\mathbf{q}}_0} & -\mathbf{W}_h\mathbf{K}_{\dot{\mathbf{q}}_f} \end{bmatrix} \Delta\mathbf{x} \\ + \quad & \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{H}_o(\mathbf{x}_d) \end{bmatrix} \end{aligned} \tag{4.23}$$

Inspection of equation 4.23 shows that the design of the feedback gains matrix

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_{\mathbf{q}_0} & \mathbf{K}_{\mathbf{q}_f} & \mathbf{K}_{\dot{\mathbf{q}}_0} & \mathbf{K}_{\dot{\mathbf{q}}_f} \end{bmatrix}$$

can be made by considering the following linear time-varying tracking error system:

$$\Delta\dot{\mathbf{x}} = \mathbf{A}(\mathbf{x}_d)\Delta\mathbf{x} + \mathbf{B}_(\mathbf{x}_d)\mathbf{e} + \delta(\mathbf{x}_d), \tag{4.24}$$

where

$$\mathbf{A}(\mathbf{x}_d) = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{H}_{\mathbf{q}_0}(\mathbf{x}_d) & \mathbf{H}_{\mathbf{q}_f}(\mathbf{x}_d) & \mathbf{H}_{\dot{\mathbf{q}}_0}(\mathbf{x}_d) & \mathbf{H}_{\dot{\mathbf{q}}_f}(\mathbf{x}_d) \end{bmatrix} \tag{4.25}$$

$$\mathbf{B}(\mathbf{x}_d) = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{I} \\ \mathbf{W}_{12}^T \mathbf{W}_{11}^{-1}(\mathbf{q}_{0d}) \end{bmatrix} \tag{4.26}$$

$$\boldsymbol{\delta}(\mathbf{x}_d) = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{I} \\ \mathbf{H}_o(\mathbf{x}_d) \end{bmatrix} \tag{4.27}$$

If $\boldsymbol{\delta}(\mathbf{x}_d)$ is considered as an uncontrollable disturbance, a linear time-varying state feedback:

$$\mathbf{e} = -\mathbf{K}\Delta\mathbf{x},$$

applied to the system described by equations 4.24, will solve the control problem of the system described by equations 4.10.

4.5.5 Compensation of static and dynamic deflection

In the control the tracking error $\Delta\mathbf{x} = \mathbf{x} - \mathbf{x}_d$ is fed back. Therefore the reference values of all state variables

$$\mathbf{x}_d = [\mathbf{q}_{0d}\ \mathbf{q}_{fd}\ \dot{\mathbf{q}}_{0d}\ \dot{\mathbf{q}}_{fd}]$$

have to be calculated.

Given is the desired trajectory of the tip. The first step is to transform the trajectory of the tip to angular positions, velocities and accelerations of virtual rigid links; $\boldsymbol{\theta}_r$, $\dot{\boldsymbol{\theta}}_r$ and $\ddot{\boldsymbol{\theta}}_r$. This is an inverse kinematic problem, for which a method is described in appendix C.

The reference values of the state variables are calculated, based on the model of the flexible robot, so that they give the same tip position trajectory as the virtual rigid link positions.

By this calculation of the reference values, based on the model of the flexible robot, static and dynamic deflection is compensated. This improves the tracking of a reference trajectory.

Calculation of the reference values will be described in more detail in chapter 5 " Practical implementation issues".

4.6 Conclusion

A nonlinear control method, applicable to multi–link flexible robots, is proposed. The method is called generalized nonlinear decoupling control. The method consists of

- a nonlinear feedforward or feedback to cancel out nonlinear terms in the rigid body dynamics.
- linearization of the remaining nonlinearities, and linear feedback of all state variables (rigid and flexible coordinates). This ensures proper stability of the system about the desired state.
- compensation of static and dynamic deflection. This improves the tracking of a reference trajectory. The compensation is achieved by calculation of the reference values of all state variables based on the model of the flexible robot.

The method is applicable to flexible link as well as to flexible joint robots. The method will be applied to a simulation model of a two–flexible link robot in chapter 7. Experimental results will be given in chapter 8 and 9. In chapter 8 the generalized nonlinear decoupling method will be applied to a two–link test set up with a flexible first joint. In chapter 9 the generalized nonlinear decoupling method will be applied to an industrial KUKA IR 161/60 robot.

The compensation torque can be a nonlinear feedforward torque or a nonlinear feedback torque. Using the feedforward dynamics compensation may have some implementational advantages. Therefore in the experiments nonlinear feedforward compensation will be applied.

To implement the generalized nonlinear decoupling controller on experimental test setups, some problems still need to be solved. This will be considered in the next chapter. The choice of sensors, estimation of velocities by a nonlinear state estimator or by differentiation, low pass filtering of measured signals and inclusion of the filter in the model, calculation of reference values for the state variables, the choice of the order of polynomials used in the trajectory generation, compensation of nonlinear Coulomb friction and the total control implementation, will be treated.

Chapter 5

Practical implementation issues

5.1 Introduction

Before implementing the generalized nonlinear decoupling controller on experimental test set ups, some practical problems have to be adressed.

Section 5.2 considers the choice of sensors used in the experiments. Motor and link positions, and flexible coordinates can be measured directly. The remaining state variables (motor and link velocities, and derivatives of flexible coordinates) need to be estimated.

Section 5.3 treats the estimation of the velocities. Two methods are proposed. The first method makes use of a nonlinear state estimator. The second method estimates the velocities by direct numerical differentiation. The first method is generally applicable to all classes of systems and has the advantage of being less sensitive to high frequency noise. The second method has the advantage of needing less computation time.

If the velocities are calculated by direct differentiation, it is necesarry to filter out high frequencies in the measured signals by low pass filters. This is also useful to overcome the problem of observation spill–over.

Section 5.3.3 describes the filtering of the measurements and the extension of the model to account for the filter dynamics.

In the control design the desired values of all state variables are

needed. Section 5.4 describes how the desired state variables are calculated given the desired tip position trajectory. Also the choice of the order of polynomials used in the trajectory generation is treated.

By compensation of the nonlinear Coulomb friction its adverse effects on the controller performance are partially eliminated. This is described in section 5.5.

A last section describes how all the different parts of the controller are organized in the implementation of the total controller.

5.2 Choice of sensors

The ideal sensor for endpoint trajectory control is an endpoint position sensor. A Rodym measurement system, consisting of a digitizer pen and tablet [76] is available. Because of the limited working area of the sensor, the digitizer tablet is not used in control. But the tablet is used to evaluate the endpoint tracking accuracy of the controllers.

In the experiments encoders are used to measure the motor position. For the flexible joint robots, additional sensors measure the deflections at the joints. For the two link test set up the joint flexibility is measured using a LVDT (linear variable differential transformer) which gives directly the difference between motor and link position. For the KUKA robot an additional encoder is mounted on the links. The deflection is then calculated as the difference between the measured motor and link position.

For the case of rigid links, from this set of sensor measurements the tip position of each link i can be derived. The tip position, y_{ei} is defined as

$$y_{ei} = l_i \theta_{li} = l_i(\theta_{mi} + q_{ji})$$

with

θ_{li} the angular position of the ith link
θ_{mi} the angular position of the ith actuator rotor
q_{ji} the ith joint deflection
l_i the length of the ith link

For the one–flexible link test set up, in addition to the motor encoders, strain gauges are used to measure the link deflection. In the experiments only the first flexible mode is considered. The strain gauges are

placed on the link position where the highest deflection occurs. The strain gauge signal is proportional to the bending

$$w''(x,t) = \Phi''(x)q_l(t) \tag{5.1}$$

where $\Phi''(x)$ is the second position derivative of the shape function $\Phi(x)$. Hence, the flexible coordinate is proportional to the measured strain gauge signal. The endpoint position is obtained using an experimentally identified linear relationship between endpoint position, motor position and strain gauge signal (see chapter 6).

In general, in case of multi–link flexible manipulators and when taking into account more than one flexible mode per link, more sets of strain gauges have to be placed in different positions on the links. The flexible coordinates associated with the ith link, are then calculated using the set of algebraic equations describing the bending in the different positions x_k.

$$w_i''(x_k,t) = \sum_{j=1}^{n_i} \Phi_{i,j}''(x_k)q_{l\,i,j}(t) \qquad k = 1, n_i \tag{5.2}$$

The tip position of each link i is given by

$$y_{ei} = l_i\theta_{li} + \sum_{j=1}^{n_i} \Phi_{i,j}(l_i)q_{l\,i,j}(t)$$

As a conclusion, motor positions and joint deflections are measured directly, and the flexible coordinates of the links are obtained from strain gauge measurements. The state variables which can not be measured directly with the available sensors are the motor angular velocities and the velocities of deflection.

5.3 State estimation

In control theory it is assumed that all the states are available for feedback. However in practice, all states can not be measured directly. The approach of state estimation is to construct an approximation of the full state vector on the basis of available measurements.

For linear structures, as the one flexible link robot, a classical linear state estimator is used for this purpose (see chapter 3). For nonlinear structures a discrete nonlinear state estimator is proposed in

subsection 5.3.1. The estimator structure is similar to a linear current estimator, but the nonlinearity of the model is taken into account. A similar solution for the estimation of states of nonlinear structures is found by Hollars [37].

In the experiments, the only states needed for feedback, which can not be measured directly, are the motor velocities and the derivatives of the flexible coordinates.

An alternative solution consists of direct numerical differentiation. This is described in subsection 5.3.2. In principle, direct differentiation is difficult to implement and has to be avoided. The reason is that differentiation is a less stable algorithm which is very sensitive to high frequency noise. In spite of this, to avoid the high computation time of the nonlinear state estimator, it can be suitable to use differentiation. To overcome the problem of sensitivity to high frequency noise, the high frequencies in the measured signals are filtered out using low pass filters. A filter introduces a phase lag. Therefore it is necessary to use exactly known digital filters and to include these filters in the model. This is described in subsection 5.3.3. Using low pass filtering is also a solution to avoid observation spill–over.

In subsection 5.3.4, the nonlinear state estimator and the calculation of velocities by differentiation are compared with respect to computation time.

5.3.1 Nonlinear state estimator

Because of the nonlinearity of the model of multi–link flexible robots, a classical linear state estimator can not be used. A solution is to linearize the model, but this is only correct in the neighbourhood of the linearization point. A better solution is to calculate a time update of the states by integration of the full non linear model.

The non linear model in state space description is given by:

$$\begin{aligned} \dot{\mathbf{x}} &= f(\mathbf{x}, \mathbf{T}) \qquad (5.3) \\ &= \begin{bmatrix} \dot{\mathbf{q}}_0 \\ \dot{\mathbf{q}}_f \\ -\mathbf{W}_{11}(\theta_l)\,\mathbf{F}_{\mathbf{q}_0}(\mathbf{x}) - \mathbf{W}_{12}(\theta_l)\,\mathbf{F}_{\mathbf{q}_f}(\mathbf{x}) \\ -\mathbf{W}_{12}^T(\theta_l)\,\mathbf{F}_{\mathbf{q}_0}(\mathbf{x}) - \mathbf{W}_{22}(\theta_l)\,\mathbf{F}_{\mathbf{q}_f}(\mathbf{x}) \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{W}_{11}(\theta_l) \\ \mathbf{W}_{12}^T(\theta_l) \end{bmatrix} \mathbf{T} \\ \mathbf{y} &= \mathbf{C}\mathbf{x} \end{aligned}$$

where $\mathbf{x}$ is the state vector, $\mathbf{y}$ is the output vector and $\mathbf{T}$ is the input

vector or vector of motor torques.

The discrete non linear state estimator is a current estimator consisting of two stages: a measurement update:

$$\hat{\mathbf{x}}_k = \tilde{\mathbf{x}}_k + \mathbf{L}(\mathbf{y}_k - \tilde{\mathbf{y}}_k)$$

and a time update:

$$\begin{aligned} \tilde{\mathbf{x}}_{k+1} &= \int_{t_k}^{t_{k+1}} f(\hat{\mathbf{x}}_k, \mathbf{T}_k)dt \\ \tilde{\mathbf{y}}_{k+1} &= \mathbf{C}\tilde{\mathbf{x}}_{k+1} \end{aligned}$$

where $f(\hat{\mathbf{x}_k}, \mathbf{T}_k)$ is the function $f(\mathbf{x}, \mathbf{T})$ (equation 5.3) evaluated at the time instant k.

The unstable differential operation needed for calculation of velocities is thus avoided and replaced by a more stable integration. The estimator structure is similar to that of a linear current estimator. The only difference is that the time update of the states is provided by integration of the full non linear model.

A simple first order Euler integrator is selected in the experiments to integrate the nonlinear equations of motion numerically. This integrator is chosen because it is easy to implement and needs little computation time. With this integrator controller stability is provided in the tests. For other set ups it can be necessary to use more complex integrating algorithms.

The estimator gain matrix, $\mathbf{L}$, is calculated using a pole placement algorithm, based on the linearized and discretized model. The model is linearized about zero velocity. In the experiments (chapter 8), only one constant feedback gain matrix is chosen: the feedback matrix is designed with respect to the final desired position.

5.3.2 Estimation of velocities by differentiation.

The nonlinear state estimator has the disadvantage of needing a large computation time. Taking this in consideration, in spite of the sensitivity to high frequency noise, it can be more suitable to calculate the velocities by differentiation. To overcome the problem of sensitivity to high frequency noise of the differentiator, the measured signals are filtered using digital low pass filters. This increases again the

computation time but in practice (for industrial robots or distributed parameter systems like flexible link robots) low pass filtering is already needed to reduce the problem of observation spillover. Hence, adding filters is not an additional cost in terms of computation time.

Differentiation is sensitive to high frequency noise, because high frequencies in the signals are strongly amplified. For example suppose the measured signal is perturbed by a high frequency signal of f_p Hz. After differentiation the differentiated signal is perturbed by a high frequency signal of the same frequency but with an amplitude which is $2\pi f_p$ times higher than the amplitude of the original perturbation. For higher (nth) derivatives, it is still worse. The amplitude of the perturbation is amplified with a factor $(2\pi f_p)^n$.

In the experiments a simple first order, backward Euler differentiation rule is used.

$$\dot{y}_k = \frac{y_k - y_{k-1}}{T_s} \tag{5.4}$$

with T_s the sample period. This differentiation rule is chosen because of the limited computation time. It is supposed that the error caused by differentiation is negligible in comparison with other errors. In the experiments good controller performance is obtained using this first order rule. However it is always possible to use higher order rules, which will reduce differentiation errors but will increase the computation time.

In chapter 8, "Experimental results on a two–link robot with a flexible first joint", estimation of velocities by a nonlinear state estimator and estimation of velocities by differentiation are compared experimentally (section 8.4.1). Similar results are obtained.

5.3.3 Filtering of the positions and extension of the model

The measured positions are filtered using low pass filters. This makes it possible to estimate the velocities by simple differentiation, which reduces the computation time in comparison with nonlinear state estimation. Using low pass filtering is also a solution to observation spill–over.

Exactly known digital filters are used. In the experiments second order Butterworth filters are used. Using higher order filters will increase the performance, but the computation time will increase considerably. (This is caused by the increased calculation time of filtering

the outputs and by the higher order of the extended model.) The breakpoints of the filter are chosen somewhat higher than the highest frequency of interest in the system. If only one flexible mode is taken into account this is given by the first resonance frequency.

A digital second order Butterworth filter is given by:

$$H_f(z^{-1}) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} \tag{5.5}$$

The filtered outputs, $\mathbf{y}_f$, are calculated as:

$$\mathbf{y}_{f,k} = b_0 \mathbf{y}_k + b_1 \mathbf{y}_{k-1} + b_2 \mathbf{y}_{k-2} - a_1 \mathbf{y}_{f,k-1} - a_2 \mathbf{y}_{f,k-2}$$

These filters must be included in the model to account for their phase lag. If a second order filter is used the order of the model is extended with order two (per included filter model). To calculate the feedback gains the linearized model of the system is needed. The extended model is then a series connection of the linearized robot model and the filter model.

Two solutions are possible. This is shown in figure 5.1.

1. In practice the outputs of the system are filtered. So, the filter model has to be put after the robot model.

2. Suppose that all outputs are filtered with the same filter. Mathematically, the same input/output behaviour is obtained if the filter is put first. In this case, the motor inputs are filtered before they are applied to the robot model.

The second solution is chosen for two reasons:

- The model is extended by a lower order, because fewer filter models have to be placed in series. The outputs of the model are the filtered positions and the filtered velocities. (Calculation of velocities by differentiation of filtered positions gives as output filtered velocities.)

  If the filter models are put after the robot model, the model needs to be extended with 2(N+P) filter models, with N the number of rigid body coordinates and P the number of flexible coordinates. The order of the model increases with 4(N+P).

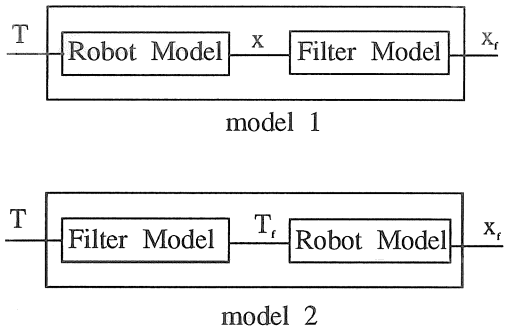

Figure 5.1: extended dynamic model.

If the filter models are put first, the model needs only to be extended with N filter models (the number of inputs). The order of the model increases with 2N.

- The state variables of the robot submodel are filtered positions and velocities in the second solution. This is not the case in the first solution. If the purpose of adding filters is to calculate the velocities from the filtered positions the second solution has to be chosen.

The inclusion of the filter in the model works as follows: The second order Butterworth filter is transformed into its state space representation, $\mathbf{F}_f, \mathbf{G}_f, \mathbf{H}_f, \mathbf{J}_f$, (controller canonical form):

$$\mathbf{F}_f = \begin{bmatrix} -a_1 & -a_2 \\ 1 & 0 \end{bmatrix} \qquad \mathbf{G}_f = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$\mathbf{H}_f = \begin{bmatrix} b_1 - a_1 b_0 & b_2 - a_2 b_0 \end{bmatrix} \qquad \mathbf{J}_f = [b_0] \tag{5.6}$$

As many filter models as there are inputs are put in parallel. For a two link robot this is:

$$\mathbf{F}_{fp} = \begin{bmatrix} \mathbf{F}_f & \mathbf{0}\,(2 \times 2) \\ \mathbf{0}\,(2 \times 2) & \mathbf{F}_f \end{bmatrix} \qquad \mathbf{G}_{fp} = \begin{bmatrix} \mathbf{G}_f & \mathbf{0}\,(2 \times 1) \\ \mathbf{0}\,(2 \times 1) & \mathbf{G}_f \end{bmatrix}$$

$$\mathbf{H}_{fp} = \begin{bmatrix} \mathbf{H}_f & \mathbf{0}\,(1 \times 2) \\ \mathbf{0}\,(1 \times 2) & \mathbf{H}_f \end{bmatrix} \qquad \mathbf{J}_{fp} = \begin{bmatrix} \mathbf{J}_f & \mathbf{0}\,(1 \times 1) \\ \mathbf{0}\,(1 \times 1) & \mathbf{J}_f \end{bmatrix} \tag{5.7}$$

Then the filter models are put in series with the linearized robot model:

$$\mathbf{F}_t = \begin{bmatrix} \mathbf{F}_{fp} & \mathbf{0} \\ \mathbf{G}_l\mathbf{H}_{fp} & \mathbf{F}_l \end{bmatrix} \qquad \mathbf{G}_t = \begin{bmatrix} \mathbf{G}_{fp} \\ \mathbf{G}_l\mathbf{J}_{fp} \end{bmatrix}$$

$$\mathbf{H}_t = \begin{bmatrix} \mathbf{J}_l\mathbf{H}_{fp} & \mathbf{H}_l \end{bmatrix} \qquad \mathbf{J}_t = [\mathbf{J}_l\mathbf{J}_{fp}] \tag{5.8}$$

with $\mathbf{F}_l, \mathbf{G}_l, \mathbf{H}_l, \mathbf{J}_l$ the state space representation in discrete time of the linearized robot model.

The state variables of this extended model are given by:

$$[\mathbf{x}_{fp} \;\; \mathbf{q}_{0f} \;\; \mathbf{q}_{ff} \;\; \dot{\mathbf{q}}_{0f} \;\; \dot{\mathbf{q}}_{ff}]^T$$

where $\mathbf{x}_{fp}$ is the state vector of the filter model 5.7, $\mathbf{q}_{0f}$ are filtered values of $\mathbf{q}_0$ and $\mathbf{q}_{ff}$ are filtered values of $\mathbf{q}_f$.

The digital filter model is linear and exactly known. Hence, by using the extended model the phase lag of the filters is taken into account and will not cause stability problems.

For state feedback, an estimate of the states of the filter is needed. These variables are calculated from the exactly known filter model:

$$\mathbf{x}_{fp,k+1} = \mathbf{F}_{fp}\,\mathbf{x}_{fp,k} + \mathbf{G}_{fp}\,\mathbf{T} \tag{5.9}$$

with initial values equal to the steady state values:

$$\mathbf{x}_{f,0} = \begin{bmatrix} \frac{1}{1+a_1+a_2} & 0 \\ \frac{1}{1+a_1+a_2} & 0 \\ 0 & \frac{1}{1+a_1+a_2} \\ 0 & \frac{1}{1+a_1+a_2} \end{bmatrix} \mathbf{T}_0$$

where $\mathbf{T}_0 = [T_{1,0} \;\; T_{2,0}]^T$ are the initial values of the torque, calculated as the torque needed to hold the system in steady state (i.e. the torque needed to compensate the gravity).

5.3.4 Comparison of the calculation of velocities by differentiation and the nonlinear state estimator with respect to computation time

To validate the statement that calculation of the velocities by differentiation decreases the computation time, the number of operations are counted for a two link robot with two flexible joints (for example the KUKA IR161/61 industrial robot described in chapter 9):

- Nonlinear state estimator
  - number of multiplications: 77
  - number of additions: 76
  - number of cosines/sines evaluations: 4
- Differentiation (first order rule)
  - number of multiplications: 4
  - number of additions: 4
- Filtering of outputs and additional computation time caused by using the extended model
  - number of multiplications: 32
  - number of additions: 30

In this calculation, the computation time is optimized by taking into account the zeros and constant values in the matrix multiplications, and by the definition of auxiliary variables for values which have to be calculated more than once.

The computation time is much smaller for the direct differentiation than for the nonlinear state estimator. Adding the second order digital low pass filters increases again the computation time, but it remains lower than one half of the computation time needed for the nonlinear state estimator. Because often, low pass filters are already needed to reduce the problem of observation spillover, they can not be considered as an additional cost in computation time.

5.4 Calculation of the desired states.

In the control the tracking error $\Delta\mathbf{x} = \mathbf{x} - \mathbf{x}_d$ is fed back. Therefore the reference values of all state variables

$$\mathbf{x}_d = [\mathbf{q}_{0d}\ \mathbf{q}_{fd}\ \dot{\mathbf{q}}_{0d}\ \dot{\mathbf{q}}_{fd}]$$

have to be calculated.

Given is the desired trajectory of the tip. The first step is to transform the trajectory of the tip to angular positions, velocities and

accelerations of virtual rigid links; θ_r, $\dot{\theta}_r$ and $\ddot{\theta}_r$. This is an inverse kinematic problem, for which a method is described in appendix C.

The reference values of the state variables are calculated so that they give the same tip position trajectory as the virtual rigid link positions. To this end the relationship between the virtual angular positions and the reference values of the motor positions and/or link positions and flexible coordinates is used. The reference values of motor positions and/or link positions and flexible coordinates are calculated based on the model of the flexible robot, taking into account this relationship.

For a flexible link robot, the computation of the desired states is an order of magnitude more complex and intricate than for a flexible joint robot. Therefore the trajectory generation for flexible joint and link robots are treated separately in subsections 5.4.1 and 5.4.2.

The derivatives are calculated using a differentiation algorithm. Because the smooth trajectories do not contain high frequency components, differentiation is no problem here. Because this is an off–line calculation, in most cases the computation time is less important. Therefore it can be suitable to choose higher order differentiation rules for these calculations, to reduce the differentiation error (see section 5.3.2).

Subsection 5.4.3 treats the problem of chosing the order of the polynomials used to generate the tip position trajectory. The reference values of the state variables are also needed for the off–line calculation of the feedforward torques. It will be shown that to avoid discontinuities in the feedforward torques, a trajectory based on a ninth order polynomial is needed.

5.4.1 Calculation of the desired state variables for a flexible link robot

For a flexible link robot the reference values of the state variables are

$$\mathbf{x}_d = [\boldsymbol{\theta}_{ld}\ \mathbf{q}_{ld}\ \dot{\boldsymbol{\theta}}_{ld}\ \dot{\mathbf{q}}_{ld}]$$

where $\boldsymbol{\theta}_{ld}$ is the vector of desired link positions and $\mathbf{q}_{ld}$ is the vector of desired generalized coordinates associated with the link flexibility.

The reference values of the state variables are calculated such that they give the same tip position trajectory as the virtual rigid links.

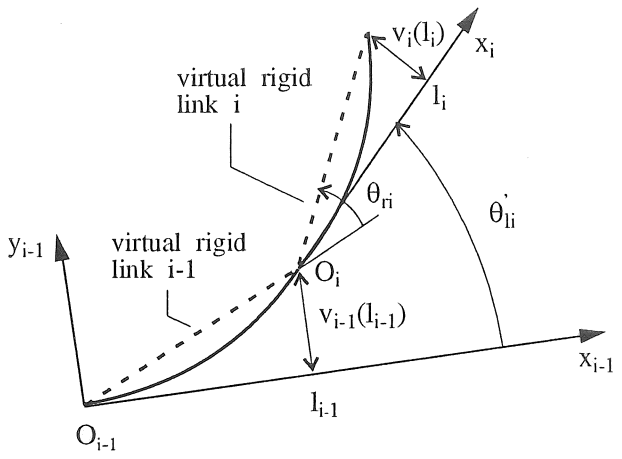

Figure 5.2: Position of virtual rigid links with respect to the tangent coordinate system

Figure 5.2 shows the relationship between the virtual angular positions, θ_{ri}, and the desired link positions, θ_{ldi}. The following relationship between the virtual angular positions and the reference values of the state variables is obtained.

$$\theta_{li} = \theta_{ri} + \frac{v_{i-1}(l_{i-1})}{l_{i-1}} - \frac{\delta v_{i-1}(l_{i-1})}{\delta x_{i-1}} - \frac{v_i(l_i)}{l_i} \tag{5.10}$$

where

$$v_i = \sum_{j=1}^{n_i} \Phi_{i,j}(l_i) q_{l\,i,j}(t)$$

In this equation the angle between the tangential coordinate axis $O_i x_i$ and the virtual rigid link i is approximated by

$$\frac{v_i(l_i)}{l_i} \tag{5.11}$$

Equation 5.10 is found by considering the angle between tangential coordinate axis $O_{i-1}x_{i-1}$ and the virtual rigid link i. This is equal to:

1. The sum of the angle between coordinate axis $O_{i-1}x_{i-1}$ and O_ix_i (equation 2.2) and the angle between coordinate axis O_ix_i and the virtual rigid link i:

$$\theta'_{li} + \frac{v_i(l_i)}{l_i} = \theta_{li} + \frac{\delta v_{i-1}(l_{i-1})}{\delta x_{i-1}} + \frac{v_i(l_i)}{l_i} \tag{5.12}$$

2. The sum of the angle between coordinate axis $O_{i-1}x_{i-1}$ and the virtual link $i-1$, and the angle between virtual link $i-1$ and virtual link i:

$$\frac{v_{i-1}(l_{i-1})}{l_{i-1}} + \theta_{ri} \tag{5.13}$$

From equation 5.12 and 5.13 follows the relationship 5.10.

Hence, the problem is reduced to the calculation of the reference values of the flexible coordinates, given the virtual link positions, θ_r. The real link positions and motor positions can then be calculated from equation 5.10.

The reference values of the flexible coordinates are calculated based on the model of the flexible robot. The model of a flexible link robot is given by equation 2.16. Consider the equations describing the dynamic behaviour of the flexible coordinates

$$\mathbf{M}^T_{rfl}(\boldsymbol{\theta}_l)\ddot{\boldsymbol{\theta}}_l + \mathbf{M}_{fl}(\boldsymbol{\theta}_l)\ddot{\mathbf{q}}_l + \mathbf{F}_{fl}(\boldsymbol{\theta}_l, \mathbf{q}_l, \dot{\boldsymbol{\theta}}_l, \dot{\mathbf{q}}_l) = \mathbf{0} \tag{5.14}$$

or

$$\mathbf{M}^T_{rfl}(\boldsymbol{\theta}_l)\ddot{\boldsymbol{\theta}}_l + \mathbf{M}_{fl}(\boldsymbol{\theta}_l)\ddot{\mathbf{q}}_l + \mathbf{C}_{fl}(\boldsymbol{\theta}_l, \dot{\boldsymbol{\theta}}_l, \dot{\mathbf{q}}_l) + \mathbf{G}_{fl}(\boldsymbol{\theta}_l, \mathbf{q}_l) + \mathbf{K}_{fl}\mathbf{q}_l + \mathbf{D}_{fl}\dot{\mathbf{q}}_l = \mathbf{0} \tag{5.15}$$

where (see chapter 2):
$\mathbf{C}_{fl}(\boldsymbol{\theta}_l, \dot{\boldsymbol{\theta}}_l, \dot{\mathbf{q}}_l)$ is the nonlinear centrifugal and coriolus terms vector,
$\mathbf{G}_{fl}(\boldsymbol{\theta}_l, \mathbf{q}_l)$ is the gravity vector,
$\mathbf{K}_{fl} = Diag\{k_{i,j}\}$ is the modal generalized stiffness matrix and,
$\mathbf{D}_{fl} = Diag\{d_{i,j}\}$ is the structural damping.

Theoretically equation 5.10 and 5.15 can be solved exactly to obtain the desired trajectories, but the equations are highly coupled and nonlinear, and difficult to solve. It is impractical to solve such complex equations numerically. In addition, solving the equations exactly will result in oscillating desired states (excitation of resonance frequencies). Using this in the controller will give oscillating torques resulting in undesired vibrations. This is what occured in the application of

the general nonlinear decoupling theory to a one flexible link robot as described in section 4.5.

For these reasons, the following approximations are made:

1. the dependency of the gravity vector on the flexible coordinates is neglected. (This is a good approximation which simplifies the calculations.)

2. the first and second derivatives of the flexible coordinates are put equal to zero.

3. in a first step, the motor positions, velocities and accelerations are taken equal to the virtual motor positions, velocities and accelerations. Once an initial estimate for the flexible coordinates is obtained, the calculations can be repeated with the corrected motor positions obtained from equation 5.10.

Making the second approximation means that only static (gravitational), inertial, centrifugal and coriolis forces are taken into account. The coupling between flexible coordinates is neglected. In practice this gives a good approximation because the main part of the deflections is introduced by these forces. In the sequel, the deflection due to gravitational forces is called static deflection and the deflection due to inertial, centrifugal and coriolis forces is called quasi–static deflection. This calculation corresponds to the calculation of the quasi-static values, as described in the calculation of desired states in linear control design (section 3.4.2.3).

The obtained equations are of the form

$$\mathbf{q}_{ld} = \mathbf{K}_{fl}^{-1} \left[-\mathbf{M}_{rfl}^{T}(\theta_r)\ddot{\theta}_r - \hat{\mathbf{C}}_{fl}(\theta_r, \dot{\theta}_r) - \hat{\mathbf{G}}_{fl}(\theta_r) \right] \tag{5.16}$$

where :
$\hat{\mathbf{C}}_{fl}(\theta_r, \dot{\theta}_r)$ is a vector containing the nonlinear centrifugal and coriolis terms, where the dependency on flexible coordinates is neglected, and $\hat{\mathbf{G}}_{fl}(\theta_r)$ is the gravity vector where the dependency on flexible coordinates is neglected.

This procedure is used in the simulation of the control of a two link flexible robot in chapter 7.

5.4.2 Calculation of the desired state variables for a flexible joint robot

For a robot with flexible joints, in addition to the reference values of the link positions and the flexible coordinates associated with the link flexibility, the reference values of the motor positions, θ_{md}, and the reference values of the flexible coordinates associated with joint flexibility, $\mathbf{q}_{jd}$, need to be calculated.

The relationship between motor position and link position is given by:

$$\theta_{m\,id} = \theta_{l\,id} - q_{j\,id} \tag{5.17}$$

Suppose the reference value of the angular link position, θ_{ld}, and the flexible coordinates associated with the link flexibility, $\mathbf{q}_{ld}$ are known. (If the robot has rigid links, the virtual angular link position, θ_r, is the reference value of the link position θ_{ld}. If the links are flexible the link positions θ_{ld} and the flexible coordinates associated with the link flexibility, $\mathbf{q}_{ld}$, are first calculated with the method explained in the previous paragraph.)

Given the link positions, velocities and accelerations and the flexible coordinates associated with the link flexibility, the flexible coordinates associated with the joint flexibility are easily calculated from the dynamic model.
Consider equation 2.17 (or 2.18):

$$(\mathbf{M}_r - \mathbf{J})\ddot{\theta}_l + \mathbf{M}_{rfl}\ddot{\mathbf{q}}_l + \mathbf{F}_r(\theta_l, \mathbf{q}_l, \dot{\theta}_l, \dot{\mathbf{q}}_l) = -\mathbf{K}_j\mathbf{q}_j$$

The desired trajectory of $\mathbf{q}_j$ is obtained by replacing in the left hand side of the equation the actual trajectories by the desired ones.

$$\mathbf{q}_{jd} = -\mathbf{K}_j^{-1}\,[\,(\mathbf{M}_r - \mathbf{J})\ddot{\theta}_{ld} + \mathbf{M}_{rfl}\ddot{\mathbf{q}}_{ld} + \mathbf{F}_r(\theta_{ld}, \mathbf{q}_{ld}, \dot{\theta}_{ld}, \dot{\mathbf{q}}_{ld})\,] \tag{5.18}$$

The desired motor position is then the difference between the desired link position and the desired deflection of the joint:

$$\theta_{md} = \theta_{ld} - \mathbf{q}_{jd}$$

The desired state trajectory calculated in this way is exact supposed that the model is exact.

5.4.3 Choice of the order of the polynomials used in trajectory generation

The reference values are needed in the feedforward calculation (equation 4.9)

$$\mathbf{T}_{ff} = -\mathbf{F}_{\mathbf{q}_0}(\mathbf{x}_d) - \mathbf{W}_{11}^{-1}\mathbf{W}_{12}(\mathbf{q}_{0d})\mathbf{F}_{\mathbf{q}}(\mathbf{x}_d) + \mathbf{W}_{11}^{-1}\ddot{\mathbf{q}}_{0d} \qquad (5.19)$$

In order to compute the feedforward signal it is necessary to differentiate the reference values of the flexible coordinates two times (this is needed for the computation of $\ddot{\mathbf{q}}_{0d}$ in equation 5.19 from equation 5.10 or 5.17). As explained in subsection 5.4.1 and 5.4.2, in order to compute the desired flexible coordinates it is necessary to differentiate the virtual link positions two times (equation 5.16 or 5.18). Hence, derivatives up to order four of the virtual link position are needed.

Discontiniuties in the feedforward signal have to be avoided, in order to avoid excitation of resonance frequencies. Therefore it is necessary that the reference trajectory has continuous derivatives up to order four. A second requirement is that the motion starts and stops in standstill. (This means that the first to fourth derivative of the trajectory are zero in the inital and final point.) To meet these requirements, a trajectory based on a ninth order polynomial is used in all simulations and experiments. The generation of such trajectories is described in appendix C. If the reference trajectory meets these requirements, perfect tracking is achieved in the absence of disturbances for flexible joints robots. (Of course the effectiveness of the feedforward compensation depends on the accuracy of the dynamic model.) For flexible link robots, the tracking is only approximately perfect because of the approximations made in the desired state calculation. The simulations on a two flexible link robot (chapter 7) will show that the made approximations are justified.

Using higher order polynomials has the disadvantage that the maximum acceleration needed for a motion from a given initial to a gven final point in a given time interval increases with the order of the polynomial (appendix C). Because saturation of motors limits the maximum achievable acceleration, the minimum time, needed for a movement from initial to final point, increases. The choice of the order of the polynomial is thus a compromise between smoothness (and thus improved performance with respect to tracking accuracy and excitation of structural dynamics) and the speed of the execution.

5.5 Coulomb friction compensation

The experiments have shown that in most test setups the Coulomb friction is not negligible. Nonlinear friction has a large effect on the design of a controller. It causes limit cycles and steady state errors, which can not be corrected with plain state feedback.

In linear control design steady state errors are eliminated using integral feedback and limit cycles are avoided by restriction of the closed loop bandwidth (chapter 3). In chapter 3 also a strategy is proposed to reduce limit cycles by phasing out the feedback once the final position is reached.

However if the nonlinear friction can be identified, a large effect of the nonlinear friction can be eliminated by direct compensation. Compensation is obtained by sending a Coulomb friction canceling torque to the motors.

The total motor friction of motor i is modeled as:

$$T_{fi} = C_{mi}\dot{\theta}_{mi} + F_{mi}sgn(\dot{\theta}_{mi}) \tag{5.20}$$

where F_{mi} : Coulomb friction of motor i
C_{mi} : linear viscous friction of motor i
$sgn(\dot{\theta}_{mi})$ $= 1$ if $\dot{\theta}_{mi} > 0$
$= 0$ if $\dot{\theta}_{mi} = 0$
$= -1$ if $\dot{\theta}_{mi} < 0$

If the Coulomb friction is included in the control model, it is automatically compensated using the generalized decoupling control method (in the same way as gravitational, centrifugal and coriolis forces are compensated). However in the experiments the Coulomb friction is not included in the control model, but compensated separately. This makes it possible to estimate and compensate the Coulomb friction on line based on the measured velocities, i.e. as a nonlinear feedback compensation torque, while the nonlinear decoupling torque is calculated off line based on desired state variables, i.e. as a nonlinear feedforward compensation torque.

Accurate identification of nonlinear Coulomb friction is difficult, since the friction can be position dependent or may vary as a function of temperature or other environmental circumstances. Care must be taken such that overcompensation does not occur since this may cause

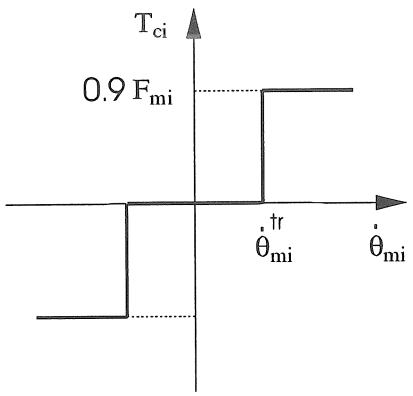

Figure 5.3: Coulomb friction canceling torque

stability problems [47]. Therefore in the experiments only 90 % of the estimated Coulomb friction is compensated.

Especially the identification of nonlinear friction around zero velocity is difficult. It is proposed in this dissertation to use a small deadband around zero velocity for the Coulomb friction canceling torque (figure 5.3). The width of the deadband around zero velocity is chosen experimentally. It is a compromise between increased steady state errors, if the width of the deadband is chosen too high, and increased amplitude of the limit cycles, if the width of the deadband is chosen too low. The increased amplitude of the limit cycles, occuring when the width of the deadband is chosen too low, are caused by small nonlinearities, like measurement noise and quantization effects, which tend to dominate the system response for motion around zero velocity [37].

The following compensation scheme for the nonlinear Coulomb friction is used (figure 5.3):

$$T_{ti} = T_i + T_{ci} \tag{5.21}$$

with

$$\begin{aligned} T_{ci} = &\quad 0.9\, F_{mi}\, sgn(\dot{\theta}_{mi}) && if \;\; |\dot{\theta}_{mi}| > \dot{\theta}^{tr}_{mi} \\ T_{ci} = &\quad 0 && if \;\; |\dot{\theta}_{mi}| \leq \dot{\theta}^{tr}_{mi} \end{aligned} \tag{5.22}$$

where $\dot{\theta}^{tr}_{mi}$ is the threshold velocity, determing the width of the deadband around zero velocity, T_{ti} is the total control torque i, T_i is the

control torque i as calculated using the generalized nonlinear decoupling theory and T_{ci} is the Coulomb friction canceling torque i.
The main effect of this compensation observed in the experiments is that the steady state errors are strongly reduced .

Because full Coulomb friction compensation is not achieved, integral action is still needed to reduce the remaining steady state error (section 3.5.1).

5.6 Total control implementation

This section summarizes the practical implementation of the generalized nonlinear decoupling controller.

The different parts as decribed in the previous sections are organized in a total control implementation in the following scheme. In the scheme the two possibilities, using a nonlinear state estimator or calculation of velocities by differentiation, are given as choice 1 and choice 2.

- OFF LINE
  - Parameter list containing:
    - model parameters
    - sample frequency
    - filter constants
    - conversion constants
    - desired closed loop control poles
    - Choice 2) desired closed loop poles for the estimator
  - Give the desired tip position trajectory
  - Calculation of virtual link trajectories (inverse kinematics)
  - Calculation of the desired state variables $\mathbf{x}_d$
  - Calculation of the feedforward torques

$$\mathbf{T}_{ff} = -\mathbf{F}_{\mathbf{q}_0}(\mathbf{x}_d) - \mathbf{W}_{11}^{-1}\mathbf{W}_{12}(\mathbf{q}_{0d})\mathbf{F}_{\mathbf{q}}(\mathbf{x}_d) + \mathbf{W}_{11}^{-1}\ddot{\mathbf{q}}_{0d}$$

  - Calculation of the desired filter state variables $\mathbf{x}_{fpd}$
    (equation 5.9 with the calculated feedforward torques $\mathbf{T}_{ff}$ as input)

- Calculation of feedback control gains, $\mathbf{K}$
  (based on the linearized and discretized model in the final desired position)
- Choice 2) Calculation of feedback estimator gains, $\mathbf{L}$
  (based on the linearized and discretized model in the final desired position)
- Initialization (initial positions, DC values, initial state estimates, etc.)

• REAL TIME CONTROL LOOP
if timer signal then

- Read in sensor data
- Run safety routine (stop if sensor data is not as expected)
- Conversion of signals to SI units
  $\mathbf{q}_{0,k}$, $\mathbf{q}_{f,k}$
- Filtering of data
  $\mathbf{q}_{0\,f,k}$, $\mathbf{q}_{f\,f,k}$
- Choice 1) Calculation of derivatives
  $\dot{\mathbf{q}}_{0\,f,k}$, $\dot{\mathbf{q}}_{f\,f,k}$
  Choice 2) Measurement update of state variables
  $\hat{\mathbf{x}}_k = \tilde{\mathbf{x}}_k + \mathbf{L}(y_k - \tilde{y}_k)$
  with $\tilde{\mathbf{x}}_k = [\tilde{\mathbf{q}}_{0f,k} \quad \tilde{\mathbf{q}}_{f\,f,k} \quad \dot{\tilde{\mathbf{q}}}_{0\,f,k}, \quad \dot{\tilde{\mathbf{q}}}_{f\,f,k}]^T$
  $\mathbf{y}_k = [\mathbf{q}_{0\,f,k} \quad \mathbf{q}_{f\,f,k}]^T$
- Calculation of integral control torque
  $\mathbf{T}_{int,k}$
- Calculation of total control torque
  $\mathbf{T}_k = \mathbf{T}_{f\,f,k} - \mathbf{K}([\hat{\mathbf{x}}_{fp,k} \quad \hat{\mathbf{x}}_k]^T - [\mathbf{x}_{fpd,k} \quad \mathbf{x}_{d,k}]^T) + \mathbf{T}_{int,k}$
- Add Coulomb friction compensation torque
  $\mathbf{T}_{t,k} = \mathbf{T}_k + 0.9\ \mathbf{F}_m\ sgn(\hat{\dot{\mathbf{q}}}_{0,k}) \qquad if \quad |\dot{\mathbf{q}}_{0,k}| \geq \dot{\theta}^{tr}_{mi}$
- Test on motor torques (for safety reasons)
- Apply calculated torques $\mathbf{T}_{t,k}$ (including Coulomb friction compensation torque)
- Estimation of next filter state variables
  $\hat{\mathbf{x}}_{fp,k+1} = \mathbf{F}_{fp}\ \hat{\mathbf{x}}_{fp,k} + \mathbf{G}_{fp}\ \mathbf{T}_k$

- Choice 2)
  - filtering of total control torque $\mathbf{T}_k$
    $\mathbf{T}_{kf}$
  - Time update of state variables
    $\tilde{\mathbf{x}}_{k+1} = \int_{t_k}^{t_{k+1}} f(\hat{\mathbf{x}}_k, \mathbf{T}_{kf})dt$

In this implementation the Coulomb friction torque is estimated and compensated on line based on the measured velocities, i.e. as a nonlinear feedback compensation torque. Because in this option the nonlinear Coulomb friction is not included in the model, in the nonlinear state estimator the torque without Coulomb friction compensation torque is used as input.

In order to minimize the time delay between the sensor readings and the command output, it is important to optimize the computation time. (This time delay is also minimized by using a current estimator instead of a prediction estimator.) The computation time is optimized by taking into account the zeros and constant values in the matrix multiplications, by eliminating loop variables, by the definition of help variables for values which have to be calculated more than once, etc.

Optimization of computation time is also important to make it possible to use reasonable sample frequencies. The choice of the sampling frequency is guided by some rules found in [7]. The selection of the sampling rate must be based on the highest frequency of interest. The highest frequency of interest depends on the aimed closed loop bandwidth. (If a state estimator is used, the highest frequency of interest depends on the bandwidth of the state estimator, which is higher than the aimed closed loop bandwidth.) Reasonable sample frequencies are 6 to 10 times higher than the highest frequency of interest.

Safety precautions are incorporated in the implementation to prevent run away motors from damaging the system. Other rules of thumb to avoid damaging the system are

- Simulate first the controller before testing it on the real test structure
- Test the state estimator separately, for example by applying bandlimited noise to the system.
- Test the program by applying first a simple PID controller instead of the generalized nonlinear decoupling controller. Com-

pare the control inputs calculated in the two different ways, compare measured and desired states, etc.

- Start with low bandwidth of the controller and increase the bandwidth until instability is observed.

5.7 Conclusion

In this chapter some problems related to practical implementation of the generalized nonlinear decoupling controller are dealt with.

The sensors used in the experiments make it possible to measure directly motor and link positions, and flexible coordinates. This chapter explaines how the remaining state variables (motor and link velocities, and derivatives of flexible coordinates) are estimated. The use of a nonlinear state estimator and the estimation of the velocities by differentiation are compared. The use of a nonlinear state estimator has the advantage of being less sensitive to high frequency noise. Estimation of velocities by differentiation has the advantage of needing less computation time.

If the velocities are calculated by differentiation, it is necesarry to filter out high frequencies of the measured signals using low pass filters. This is also useful to overcome the problem of observation spill–over. If filters are used the model has to be extended to account for the filter dynamics.

It is described how the reference values for all the state variables, needed in the control design, are calculated, given the desired tip position trajectory.

In addition a solution to overcome the problems caused by nonlinear Coulomb friction is proposed.

In summary it is described how all the different parts of the controller are organized in the implementation of the total controller.

Part II

Nonlinear Control of Flexible Robots: Simulation and Experimental Results

Chapter 6

Experimental results on a flexible one-link robot

6.1 Introduction

Identification and control of a flexible one–link robot is a first step in the development of a controller for a more complex configuration. Because the model describing the dynamics of a flexible one–link robot is linear, classical linear identification and control techniques can be used.

In this chapter a state feedback controller with acceleration feedforward for accurate tracking control of a flexible one-link robot has been implemented on a test setup. The identification and control design as described in chapter 3, "Linear identification and control of flexible robot structures", is used.

The results on a flexible one–link robot have to be considered as a first step in the development of a controller for multi–link flexible robots. The problem of linear identification and control of flexible one–link robots is treated in more detail by Swevers [69].

The model, on which the controller is based, is the result of a least squares parameter estimation algorithm. The total model is divided into two submodels in series, the parameters of which are estimated separately. The first model relates the input to the link deflection. The second model relates the link deflection to the angular motor position. The theoretical model form is derived from a physical analysis of the system.

The controller does not use a direct end point position measurement, but estimates this position from the measured motor angle and strain gauge signals. Acceleration feedforward is introduced in the state feedback controller to reduce tracking errors to negligible values. The calculation of the acceleration feedforward is based on the quasi–static behaviour of the system.

Integral control eliminates positioning errors caused by nonlinear motor friction. Limit cycles are reduced by phasing out the feedback once the final position is reached. Observation spill-over is avoided by filtering the output signals using a second order digital filter. These filters extend the model order. The feedback gains are calculated using a pole placement algorithm. The best control results are obtained for a closed loop bandwidth which is a little higher than the first resonance frequency of the structure. This controller allows accurate tracking and positioning with very limited overshoot, oscillation or tracking errors. Higher bandwidths result in limit cycles, which inhibit accurate control.

Section 6.2 gives a description of the test setup along with some dynamic characteristics. Section 6.3 describes the parameter identification. The experimental identification results are copied from [66]. Section 6.4 describes the control design, and discusses the control results.

6.2 Description of the flexible one-link robot

Figure 1 gives a schematic representation of the test setup. It consists of a flexible beam with a payload, connected to a flexible torsional beam, which itself is connected to a direct drive motor (maximum motor torque = 200 Nm). The control input to the system is a voltage between -10 Volt and $+10$ Volt, which is converted by the power supply of the motor into a current. This conversion is linear. A built-in encoder measures the angular motor position. It has a resolution of 1024000 pulses per revolution. One encoder pulse corresponds to an angle of $6.1359\,10^{-6}$ radians or 1.2656 arcseconds. Strain gauges on the flexible beam near to the axis of rotation, and in the middle of the torsional beam measure the beam deflections. The end point position is not measured directly, but can be determined from the strain gauge and encoder signals. The first resonance frequency is a free-free mode

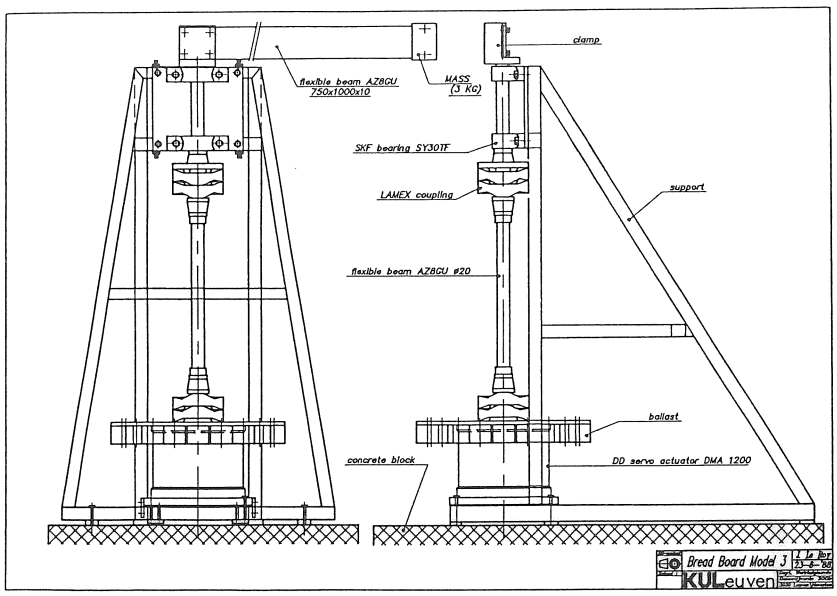

Figure 6.1: Schematic representation of the test setup.

in the neighbourhood of $5Hz$. The second resonance frequency is at $38Hz$. This mode and the higher ones have been neglected in the identification and control. An anti-resonance exists at approximately $3Hz$, which corresponds to a lowly damped clamped-free mode. This mode is excited when, in the neighbourhood of the destination point, the motor velocity suddenly drops to zero due to dry friction (static friction torque $\simeq 6\,Nm$, i.e. 3% of the maximum torque), or when external forces are applied to the beam.

6.3 Least squares identification of a flexible one-link robot

The identification of the total multi variable state space model is based on the identification of two transfer functions and on the determination of the static relation between the end point position, the motor angle and the beam deflections. The complexity of the model is determined from a physical analysis of the system. A least squares parameter estimation algorithm calculates the unknown parameters of the trans-

fer functions. These estimations are based on measured input-output signals which result from an excitation with a band limited random signal. The use of a band limited random signal reduces the unnecessary excitation of the higher modes during the measurement. The cut-off frequency of the input signal spectrum is $20Hz$. The relation between the encoder signal, the strain gauge signals and the end point position is linear if only the first mode is considered. The coefficients of this relation are determined using a static measurement.

6.3.1 Mathematical formulation of the model

The theoretical model form can be derived from a physical analysis of the system. The two flexible parts of the test setup can be modelled as a one flexible link robot, because the inertia of the torsional beam is negligible compared to the inertias of the motor on the one hand and the flexible beam with payload on the other hand. This simplification has been verified experimentally. The strain gauge signal measured on the torsional beam contains the same information as the signal of the flexible beam strain gauges. Only one of these signals has to be measured for the identification and control. The strain gauge signals of the flexible beam are chosen to represent the beam deflections. This choice was totally arbitrary. The linear motor friction is also neglected in the models. Experiments have shown the correctness of this simplification.

The theoretical model of a one flexible link robot moving in the horizontal plane is derived in chapter 3. The model is given by equation 3.4. Some interesting transfer functions can be derived from this theoretical model:

$$
\begin{aligned}
H_1(s) &= \frac{\varepsilon}{u}(s) = \frac{k_\varepsilon}{k_u}\,\frac{1}{I_t}\,\frac{\frac{d\Psi}{dx''}(0)}{s^2+2\zeta\omega s+\omega^2} \\
H_2(s) &= \frac{\theta}{\varepsilon}(s) = \frac{1}{k_\varepsilon}\,\frac{1+\frac{d\Psi}{dx''}^2(0)}{\frac{d\Psi}{dx''}(0)}\,\frac{s^2+\frac{2\zeta\omega}{1+\frac{d\Psi}{dx''}^2(0)}\,s+\frac{\omega^2}{1+\frac{d\Psi}{dx''}^2(0)}}{s^2} \\
H_3(s) &= \frac{\theta}{u}(s) = H_1(s)\;H_2(s)
\end{aligned}
\tag{6.1}
$$

$H_1(s)$ relates the input signal u to the strain gauge signal ε. $H_2(s)$ relates the strain gauge signal ε to the angular motor position θ. $H_3(s)$ relates the input signal u to the angular motor position θ. Only the

first flexible mode is considered here. k_ε is the proportional constant between flexible coordinate q and strain gauge signal ε:

$$\varepsilon = k_\varepsilon q$$

k_u is the proportional constant between input signal u and motor torque T:

$$u = k_u T$$

The discrete time transfer functions that model the test setup are the z-transform equivalents of these continuous time transfer functions.

$$\begin{aligned} H_1(z^{-1}) &= \frac{b_{11}z^{-1}}{1 + a_{11}z^{-1} + a_{12}z^{-2}} \\ H_2(z^{-1}) &= \frac{b_{20} + b_{21}z^{-1} + b_{22}z^{-2}}{1 - 2z^{-1} + z^{-2}} \\ H_3(z^{-1}) &= H_1(z^{-1})H_2(z^{-1}) \end{aligned} \tag{6.2}$$

$H_3(z^{-1})$ is not equivalent to the z-transform of $H_3(s)$. The difference between both discrete transfer functions is of second order. It has been proven experimentally that the model form $H_3(z^{-1})$ allows an accurate fitting of the data. The denominator coefficients -2 and 1 in $H_2(z^{-1})$ define a double pole at $z = 1$.

The unknown parameters $a_{11}, a_{12}, \ldots$, are calculated with a least squares time domain parameter estimation algorithm, which is explained in chapter 3. The experimental results of this identification are given in the next paragraph.

6.3.2 Least squares time domain parameter estimation

A least squares time domain parameter estimation algorithm fits a discrete domain transfer function to a given set of input-output data. This method is described in chapter 3. The data are filtered with a second order digital low pass Butterworth filter with a cut-off frequency at $10Hz$.

The total model of the test setup is based on three transfer functions, $H_1(z^{-1})$, $H_2(z^{-1})$ and $H_3(z^{-1})$. Identification of $H_1(z^{-1})$, based on the measured input and strain gauge signals, and identification of $H_3(z^{-1})$, based on the measured input and encoder signals, is the most

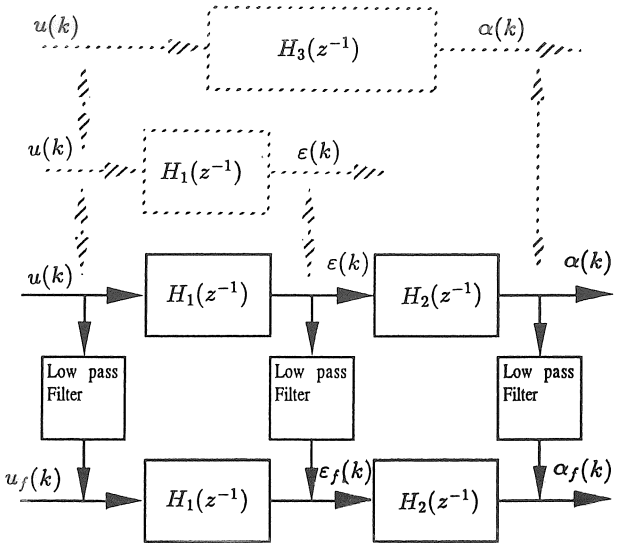

Figure 6.2: Schematic representation of the relation between $H_1(z^{-1})$, $H_2(z^{-1})$, and $H_3(z^{-1})$, used for the time domain parameter estimation.

straightforward approach, but certainly not optimal. Identification of the two second order submodels $H_1(z^{-1})$ and $H_2(z^{-1})$ is a better solution. $H_3(z^{-1})$ is calculated as a series combination of $H_1(z^{-1})$ and $H_2(z^{-1})$. Figure 3 gives the schematic representation of the division into two submodels. It also shows the low pass filtering necessary for a reliable least squares parameter estimation. Less calculation time and more accurate results are two important advantages of this method with respect to the direct estimation of the parameters of $H_3(z^{-1})$. This approach also guarantees a minimal multi variable model because it estimates every pole and zero of the transfer function only once. The eigenfrequency is estimated in $H_1(z^{-1})$. The anti–resonance frequency is estimated in $H_2(z^{-1})$. This is not the case in the straightforward approach. This approach estimates the eigenfrequency twice. It results in two different values for the same pole. The two different numerical values must be replaced by one before the controller design can be started.

The input-output signals used for the identification are measured with a sample frequency of $200Hz$. (The same sample frequency is used for the control). The identification of the parameters of $H_1(z^{-1})$

is based on the filtered input signal $u_f(k)$ and the filtered strain gauge signal $\varepsilon_f(k)$. The identification of the parameters of $H_2(z^{-1})$ is based on the filtered strain gauge signal $\varepsilon_f(k)$ and the filtered encoder signal $\theta_f(k)$. Only two times three parameters have to be estimated. b_{11}, a_{11} and a_{12} for $H_1(z^{-1})$ and b_{20}, b_{21} and b_{22} for $H_2(z^{-1})$. The a priori information of the denominator coefficients -2 and 1 of $H_2(z^{-1})$ is explicitely taken into account during the parameter estimation of the numerator coefficients of $H_2(z^{-1})$ by considering the values $\theta_f(k) - 2\theta_f(k-1) + \theta_f(k-2)$ instead of $\theta_f(k)$ as the output signal.

The obtained parameters of $H_1(z^{-1})$ and $H_2(z^{-1})$ are:

$$\begin{aligned} H_1(z^{-1}) &= \frac{-3.80z^{-1}}{1 - 1.97z^{-1} + 0.990z^{-2}} \\ H_2(z^{-1}) &= \frac{-1.42\,10^{-4} + 2.82\,10^{-4}z^{-1} - 1.42\,10^{-4}z^{-2}}{1 - 2z^{-1} + z^{-2}} \end{aligned} \tag{6.3}$$

$H_3(z^{-1})$,which is a series combination of $H_1(z^{-1})$ and $H_2(z^{-1})$, is:

$$H_3(z^{-1}) = \frac{5.40\,10^{-4}z^{-1} - 1.07\,10^{-3}z^{-2} + 5.39\,10^{-4}z^{-3}}{1 - 3.97z^{-1} + 5.92z^{-2} - 3.95z^{-3} + 0.990z^{-4}} \tag{6.4}$$

The eigenfrequency is estimated in $H_1(z^{-1})$. Its value is $4.89Hz$ with a damping ratio of 0.03 .The anti-resonance frequency is estimated in $H_2(z^{-1})$. Its value is $2.87Hz$ with a damping ratio of 0.008.

The results of the time domain identification method are compared with a non-parametric reference model. This reference model is identified using a stepped sine identification method [77]. This identification technique calculates the frequency response of the system by exciting it with constant-frequency sinusoids. Figure 4 shows the amplitude and the phase of models $H_1(z^{-1})$ and $H_3(z^{-1})$ along with the diagrams of the reference model. There is only a small difference between the diagrams in the neighbourhood of the resonance and anti-resonance frequency. This difference is caused by nonlinear motor friction [53].

6.3.3 Spill-over

In order to reduce spill-over problems the outputs are filtered with a digital low pass filter. The filters extend the continuous time physical system and the model with their discrete transfer function. The filter is the same as used for the identification. It has the following discrete

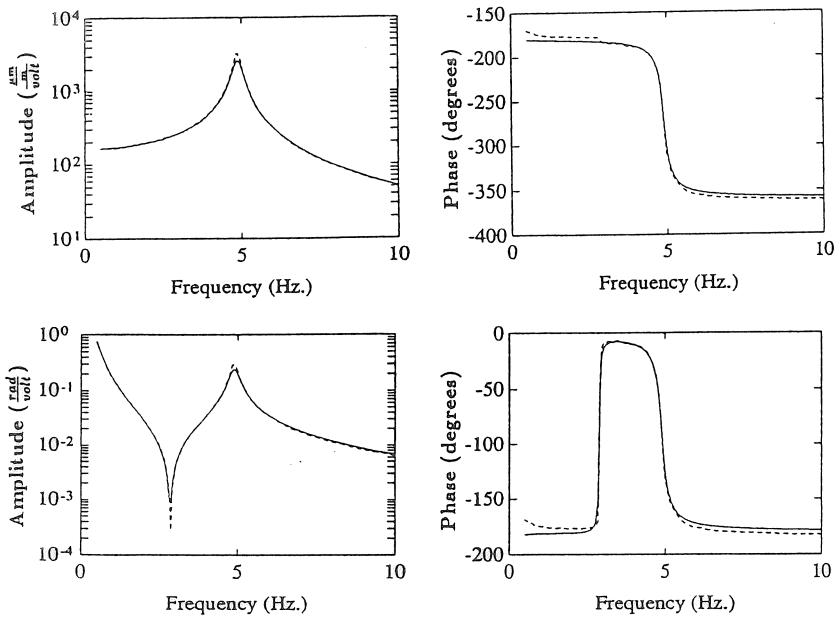

Figure 6.3: Comparison between the reference models (dotted line) and the identified transfer functions $H_1(z^{-1})$ (upper figures) and $H_3(z^{-1})$ (bottom figures) (solid line).

transfer function:

$$H_f(z^{-1}) = \frac{2.01\,10^{-2} + 4.02\,10^{-2}z^{-1} + 2.01\,10^{-2}z^{-2}}{1 - 1.56z^{-1} + 6.41z^{-2}} \quad (6.5)$$

The discrete transfer function that relates the input signal $u(k)$ to the filtered strain gauge signal $\varepsilon_f(k)$ is a series combination of $H_1(z^{-1})$ and $H_f(z^{-1})$. The discrete transfer function that relates the input signal $u(k)$ to the filtered encoder signal $\theta_f(k)$ is a series combination of $H_3(z^{-1})$ and $H_f(z^{-1})$.

6.3.4 The identification of the static relation between the end point position, the strain gauge signal and the motor angle

The position $y_e(k)$ (in mm arc length) of the end point is a linear combination of the strain gauge signal $\varepsilon(k)$ and the motor angle $\theta(k)$ if only the first mode is considered.

$$y_e(k) = a_{p_1}\varepsilon(k) + a_{p_2}\theta(k) \quad (6.6)$$

a_{p_2} is equal to the length of the flexible beam ($840\,mm$). a_{p_1} has been determined experimentally. During the experiment the end point position is held constant, while the motor angle and the beam deflection are increased. The values of the strain gauge and encoder signals are measured for different beam deflections, and used to calculate a_{p_1}. The result of this calculation is $a_{p_1} = 0.1234\ mm/\frac{\mu}{m}$. The same coefficients are valid for the relation between the filtered values of these signals.

6.3.5 The total state space model

The dynamic and static identification results (equations 6.3, 6.4 and 6.6) are then represented in a discrete state space model, in which the digital low pass filter is taken into account:

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{F}\mathbf{x}(k) + \mathbf{G}u(k) \\ \mathbf{y}(k) &= \mathbf{H}\mathbf{x}(k) \end{aligned} \quad (6.7)$$

with

$$\mathbf{x}(k) = [\ x_1(k) \quad x_2(k) \quad x_3(k) \quad x_4(k) \quad x_5(k) \quad x_6(k)\ ]^T$$

$$\mathbf{F} = \begin{bmatrix} 3.53 & -4.70 & 2.81 & -0.63 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 3.83\,10^{-5} & -7.24\,10^{-5} & 3.05\,10^{-5} & 3.94\,10^{-6} & 2 & -1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1.08\,10^{-5} & 0 \end{bmatrix}^{\mathrm{T}}$$

$$\mathbf{y}(k) = \begin{bmatrix} \varepsilon_f(k) & \theta_f(k) & y_{ef}(k) \end{bmatrix}^{\mathrm{T}}$$

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}_1 \\ \mathbf{h}_2 \\ \mathbf{h}_3 \end{bmatrix} = \begin{bmatrix} -7.63\,10^{-2} & -1.53\,10^{-1} & -7.63\,10^{-2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ -9.41\,10^{-3} & -1.88\,10^{-2} & -9.41\,10^{-3} & 0 & 8.40\,10^{2} & 0 \end{bmatrix}$$

The state space model is a combination of a canonical model and a state space model with measurable state variables. This combines the advantages of a minimum number of model parameters and the possibility to use a reduced order state estimator. The input-to-filtered strain gauge signal submodel has a controller canonical form. It is derived from the discrete transfer function that relates these two signals, and depends only on the first 4 state variables. The second submodel contains $x_5(k)$ and $x_6(k)$. $x_5(k)$ is the filtered motor position at the current time instant k, and $x_6(k)$ is the filtered motor position at the previous time instant $k-1$. The numerator coefficients of $H_2(z^{-1})$ determine the link between both submodels.

6.4 State feedback control

6.4.1 The state estimator

A reduced order current state estimator has been implemented for the test setup. State variables $x_1(k), x_2(k), x_3(k)$, and $x_4(k)$ are calculated with a closed loop estimator. The estimator is based on the canonical submodel that contains only these four state variables. It feeds back the filtered strain gauge signal $\varepsilon_f(k)$ because this is the only output signal that detects both the resonance and anti-resonance frequency. $x_5(k)$ and $x_6(k)$ are the filtered values of the measured angular motor

position. The feedback gains of the state estimator are calculated using a pole placement algorithm. The choice of the estimator poles is not so critical. The control results shown in this paper are obtained with an estimator bandwidth of $20Hz$.

6.4.2 The control design

The output of the state feedback controller consists of an acceleration feedforward and a feedback of the difference between the measured and the desired state variables:

$$u(k) = u_{ff}(k) - \sum_{i=1}^{6} K_i(x_i(k) - x_{id}(k)) \tag{6.8}$$

The control design consists of the calculation of: 1) the feedback gains K_i, 2) the feedforward term $u_{ff}(k)$, and 3) the desired state variables.

1) The feedback gains are calculated using a pole placement algorithm. A healthy approach to pole placement is to replace only the poles of the open loop system that result in a bad dynamic behaviour under tracking conditions. The two complex conjugated poles of the resonance frequency and the two poles at $z = 1$ are on or too close to the unit circle in the z-plane. They are not suited for a tracking control action and must be replaced. The system poles that originate from the Butterworth filter $H_f(z^{-1})$ result in an acceptable dynamic behaviour [27], and can be copied to the closed loop system. The choice of the remaining four pole positions is motivated by robustness considerations. Two poles are placed at $\omega_0 = 2\pi f \frac{rad}{sec}$ and $\zeta = 1.0$ for good tracking dynamics, and two poles are placed at $\omega_0 = 2\pi f \frac{rad}{sec}$ and $\zeta = 0.707$ for good regulator dynamics [23]. f is the frequency of the poles. These poles are dominant if f is smaller than the cut-off frequency of the Butterworth filter ($10Hz$). A frequency f of $7Hz$ for the dominant poles of the controller resulted in the best performance for the tracking test. The feedback gains for this controller are:

$$\mathbf{K} = \begin{bmatrix} 0.641 & -1.85 & 1.46 & -0.248 & 1.12\,10^4 & -1.06\,10^4 \end{bmatrix}$$

The closed loop bandwidth is approximately $7Hz$. The controller becomes unstable if the dominant poles are larger than $8Hz$.

2) The feedforward term is determined as being the input needed to obtain exact correspondence between output, y_e, and desired trajectory, y_{ed}. Because the calculation of the inverse model 6.7 gives stability problems, only acceleration feedforward is introduced;

$$u_{ff} = N_a \, \ddot{y}_{ed} \tag{6.9}$$

with $\ddot{y}_{ed}$ the desired acceleration.

The constant N_a is determined as:

$$\frac{1}{N_a} = \frac{\ddot{y}_e}{u_{ff}} = \frac{\ddot{y}_{ef}}{u_{ff}} = \lim_{z \to 1} (1 - z^{-1})^2 \, \mathbf{H}_3 (z\mathbf{I} - \mathbf{F})^{-1} \mathbf{G} = 0.1577 \tag{6.10}$$

The calculation of the constant N_a as the low frequency gain on the Bode diagram means that only the rigid body dynamics are considered.

3) The desired state variables x_{1d} to x_{6d} can be calculated by simulating the identified model (6.8) with input u_{ff}. Clearly, the input will excite the resonance frequency which results in an undesirable trajectory for the states. Therefore, in the calculation of desired states only rigid body dynamics and deflections due to inertial forces (quasi–static deflections) are considered.

First the desired values for the output variables ε_d and θ_d are calculated. For the calculation of ε_d only the deflection due to inertial forces is considered. Hence it appears that ε_d is proportional to the desired acceleration $\ddot{y}_{ed}$ and to the feedforward input u_{ff}. The proportional constant is calculated as (low frequency gain on the Bode diagram):

$$\frac{\varepsilon_d}{u_{ff}} = \lim_{z \to 1} H_1(z^{-1}) = 170.96 \tag{6.11}$$

The trajectory of the motor angle, θ_d, is determined by using the experimentally found relationship (6.6) between the end point position, the strain gauge signal and the motor angle:

$$\theta_d = \frac{y_{ed} - a_{p1} \varepsilon_d}{a_{p2}} \tag{6.12}$$

Second, the desired states are calculated. The desired state variables x_{5d} and x_{6d} can be calculated using the output equation of the model (6.8).

$$\begin{aligned} x_{5d}(k) &= \theta_{df}(k) \\ x_{6d}(k) &= x_{5d}(k-1) = \theta_{df}(k-1) \end{aligned} \tag{6.13}$$

where the filtered motor angle, θ_{df} is given by:

$$\theta_{df} = H_f(z^{-1})\,\theta_d \tag{6.14}$$

For the calculation of the desired state variables x_{1d} to x_{4d} the discrete transfer function $\frac{x_1}{\varepsilon_f}$ is considered. This transfer function is obtained from the output equation of the identified model (6.7).

$$\varepsilon_f(k) = -7.63\,10^{-2}x_1(k) - 1.53\,10^{-1}x_2(k) - 7.63\,10^{-2}x_3(k) \tag{6.15}$$

The model (6.7) is in canonical form, or

$$\begin{aligned} x_2(k) &= x_1(k-1) \\ x_3(k) &= x_1(k-2) \\ x_4(k) &= x_1(k-3) \end{aligned} \tag{6.16}$$

Using the backward–shift operator q^{-1} to equation 6.15 (q^{-1} is defined as $q^{-1}x_1(k) = x_1(k-1)$) gives

$$\varepsilon_f = (-7.63\,10^{-2} - 1.53\,10^{-1}q^{-1} - 7.63\,10^{-2}q^{-2})x_1 \tag{6.17}$$

The discrete transfer function $\frac{x_1}{\varepsilon_f}(z)$ is then given by

$$\frac{x_1}{\varepsilon_f}(z) = \frac{1}{-7.63\,10^{-2} - 1.53\,10^{-1}z^{-1} - 7.63\,10^{-2}z^{-2}} \tag{6.18}$$

This is the discrete transfer function of a type 0 system (no integrations). The quasi–static value of x_{1d} is proportional to ε_{fd}

$$x_{1d} = N_x\varepsilon_{fd} \tag{6.19}$$

The proportional constant N_x is calculated as (low frequency gain on the Bode diagram):

$$N_x = \lim_{z\to 1}\frac{x_1}{\varepsilon_f}(z) = -3.077 \tag{6.20}$$

The states x_{2d} to x_{4d} are equal to x_{1d} but shifted one time step.

6.4.3 Control results

A tracking test has been executed to validate the controller performance. The reference trajectory is based on a parabolic acceleration profile. The generation of such reference trajectories is described in appendix C. It gives intermediate positions between the initial and final position along a smooth path, and therefore avoids unnecessary excitation of the eigenfrequencies. Prevention is better than cure. The maximum end point acceleration and end point velocity during the tracking are 24.2 $\frac{m}{s^2}$ and 8.0 $\frac{m}{s}$. These values are near to the limitations of the motor. The total displacement of the end point is 5.28 m, which corresponds to one motor revolution. The final position should be reached after 1.34 seconds.

Integral action to reduce steady state errors and the phasing out strategy to reduce limit cycles are added. These strategies are described in chapter 3. The values of the integral feedback gain k_i, the value δ_i defining the vicinity of the final position with respect to the integral action, and the value δ_m defining the vicinity of the final position with respect to the phasing out strategy, are determined experimentally [10]. Good results are obtained with:

$k_i = 2.84 \frac{Volt}{mm\,s}$
$\delta_i = 2.5\,mm$
$\delta_m = 2.35\,rad^2$

Figure 6.4 shows the control results. There are no oscillations at a frequency equal to the resonance or anti-resonance frequency. The steady state error is equal to zero. The maximum tracking error is 11 mm.
Figure 6.5 shows the measured end point position, when the end point is in the vicinity of the final position. This position is directly measured using a digitizer board [76].
Let Δt be $t - t_{fin}$ where

t_{fin} = time at which the desired final position should be reached (1.34 s)
t = time

The following results are obtained:

- The maximum overshoot is equal to 0.42 mm and occurs at $\Delta t = 0.56\,s$.

- The position error remains within the set error tolerance of 0.5 mm after $\Delta t = 0.19\ s$.
- The amplitude of the remaining oscillation is less than 0.1 mm after $\Delta t = 0.86\ s$. This oscillation is due to the resolution of the strain gauge signal (see equation 6.6). (The influence of encoder resolution is much smaller.)

6.5 Conclusion

In this chapter a controller is designed for a flexible one–link robot based on an experimentally identified model. The control law consists of state feedback, integral feedback, and acceleration feedforward.

Nonlinear joint friction has a large effect on the design of a linear controller for a flexible robot. On the one hand it influences the accuracy of the identified linear model on wich the controller is based. On the other hand nonlinear friction causes limit cycles and steady state errors, which can not be corrected with plain state feedback. Limit cycles are avoided by a restricted closed loop bandwidth and by phasing out of the feedback once the final position is reached. Steady state errors are eliminated using integral feedback.

Large tracking errors are another defect of plain state feedback controllers. Acceleration feedforward is the right remedy for this defect. It reduces the tracking error to negligible values.

The obtained controller allows accurate tracking and positioning with very limited overshoot, oscillation or tracking errors.

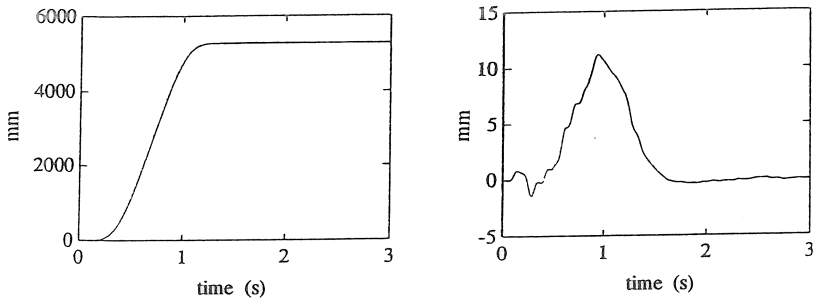

Figure 6.4: Control results. Left figure: measured (solid line) and desired (dashed line) end point position. Right figure: tracking error.

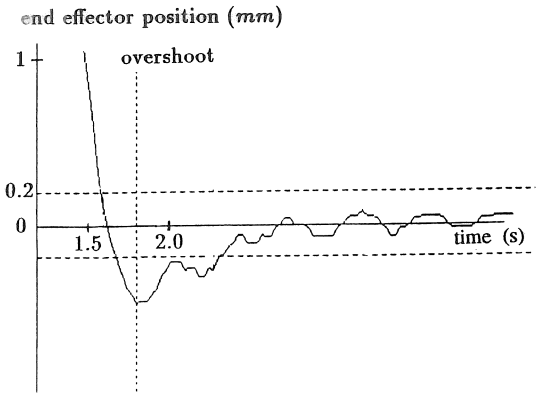

Figure 6.5: End point position in the vicinity of the final position, measured using a digitizer board.

Chapter 7

Simulation results on a two–link flexible robot

7.1 Introduction

The generalized nonlinear decoupling method is applied to a simulation model of a two–link flexible robot. Section 7.2 describes the simulation model. Section 7.3 describes the control design. Section 7.4 gives the simulation results. In comparison with a rigid controller the excited vibrations are damped out very rapidly. By taking the flexibility of the structure into account in the trajectory generation the tracking accuracy is improved considerably. Both static deflection due to gravity and dynamic deflection are compensated.

7.2 Simulation model

Figure 7.1 gives the definition of the planar two–link flexible robot moving in the vertical plane. Table 7.1 gives a list of used symbols. The nonlinear equations of motion are derived based on the Lagrangian–assumed modes formulation, as described in chapter 2.

The coordinate transformation matrices $\mathbf{C}_1$ and $\mathbf{C}_2$ (equation 2.1) are given by

$$\mathbf{C}_1 = \begin{bmatrix} cos(\theta_1) & -sin(\theta_1) \\ sin(\theta_1) & cos(\theta_1) \end{bmatrix}$$

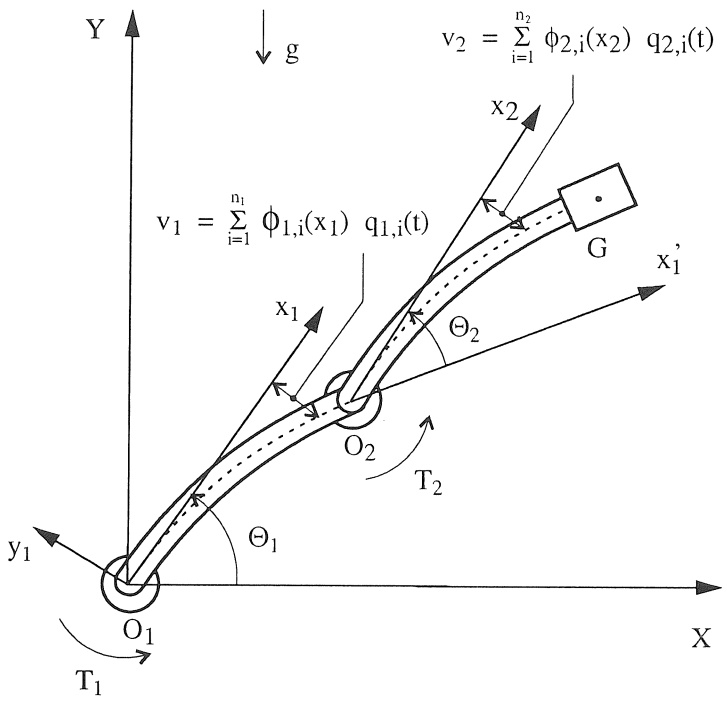

Figure 7.1: Planar two flexible link model

I_{t1}	$= I_{b1} + I_{h1}$: inertia of link and hub 1 around z trough O_1
I_{t2}	$= I_{b2} + I_{h2} + I_p + m_p l_2^2$: total inertia of link 2, hub 2 and the payload around z through O_2
m_i, l_i	mass and length of link i
m_p, I_p	mass and moment of inertia of the payload
m_{h2}	mass of joint 2
$\Phi_{1,j}(x_1)$	constrained shape functions of link 1
$\Phi_{2,k}(x_2)$	constrained shape functions of link 2
$q_{1,j}(t)$	generalized coordinates of link 1
$q_{2,k}(t)$	generalized coordinates of link 2
$\eta_{1,j}$	$= \int_0^{l_1} \rho S \Phi_{1,j}(x_1) dx_1$
$\eta_{2,k}$	$= \int_0^{l_2} \rho S \Phi_{2,k}(x_2) dx_2$
$K_{1,j}$	$= \int_0^{l_1} \rho S x_1 \Phi_{1,j}(x_1) dx_1$
$K_{2,k}$	$= \int_0^{l_2} \rho S x_2 \Phi_{2,k}(x_2) dx_2$
A_1	$= I_{t1} + I_{t2} + (m_{h2} + m_2) l_1^2 + m_p l_1^2$
A_c	$= (m_p + \frac{m_2}{2}) l_1 l_2$
$\gamma_{2,k}$	$= K_{2,k} + m_p l_2 \Phi_{2,k}(l_2) + I_p \frac{d\Phi_{2,k}}{dx_2}(l_2)$
γ_{cj}	$= (m_p + \frac{m_2}{2}) l_2 \Phi_{1,j}(l_1)$
$\mu_{2,k}$	$= \eta_{2,k} + m_p \Phi_{2,k}(l_2)$
$m_{1,j}$	$= \int_0^{l_1} \rho S \Phi_{1,j}^2(x_1) dx_1$
$m_{2,k}$	$= \int_0^{l_2} \rho S \Phi_{2,k}^2(x_2) dx_2$
$m_{1,i,j}$	$= \int_0^{l_1} \rho S \Phi_{1,i}(x_1) \Phi_{1,j}(x_1) dx_1$
$m_{2,i,j}$	$= \int_0^{l_2} \rho S \Phi_{2,i}(x_2) \Phi_{2,j}(x_2) dx_2$
$m'_{1,j}$	$= m_{1,j} + \Phi_{1,j}^2(l_1)(m_p + m_{h2} + m_2) + I_{t2} \left[\frac{d\Phi_{1,j}}{dx_1}(l_1)\right]^2$: generalized mass of jth mode of link 1
$m'_{2,k}$	$= m_{2,k} + \Phi_{2,k}^2(l_2) m_p + I_p \left[\frac{d\Phi_{2,k}}{dx_2}(l_2)\right]^2$: generalized mass of kth mode of link 2
$k_{i,j}$	$= \int_o^{l_i} (EI)_i \left[\frac{\delta^2 \Phi_{1,j}}{\delta x_i^2}\right]^2 dx_i = m'_{i,j} \omega_{i,j}^2$
$\omega_{i,j}$	jth cantilevered mode eigenpulsation of the ith link
$\zeta_{i,j}$	modal damping of the jth mode of the ith link

Table 7.1: Nomenclature

$$\mathbf{C}_2 = \begin{bmatrix} cos(\theta_1+\theta_2) & -sin(\theta_1+\theta_2) \\ sin(\theta_1+\theta_2) & cos(\theta_1+\theta_2) \end{bmatrix} + \begin{bmatrix} -sin(\theta_1+\theta_2) & -cos(\theta_1+\theta_2) \\ cos(\theta_1+\theta_2) & -sin(\theta_1+\theta_2) \end{bmatrix} \frac{\delta v_1(l_1)}{\delta x_1} \quad (7.1)$$

$\mathbf{r}_1^1, \mathbf{r}_1^2$ and $\mathbf{r}_2^2$ are given by

$$\mathbf{r}_1^1 = \begin{bmatrix} x_1 \\ \sum_{i=1}^{n_1} \Phi_{1,i}(x_1) q_{1,i}(t) \end{bmatrix} \qquad \mathbf{r}_1^2 = \begin{bmatrix} l_1 \\ \sum_{i=1}^{n_1} \Phi_{1,i}(l_1) q_{1,i}(t) \end{bmatrix}$$

$$r_2^2 = \begin{bmatrix} x_2 \\ \sum_{i=1}^{n_2} \Phi_{2,i}(x_2) q_{2,i}(t) \end{bmatrix}$$

Since the robot is moving in the vertical plane $\mathbf{g}^T = [0\ ,\ g]^T$, with g the gravitational constant.

Calculating the kinetic energy (equation 2.8), the elastic potential energy (equation 2.10) and the gravitational potential energy (equation 2.9) and taking the appropriate derivatives gives the dynamic equations. They are summarized by

$$\begin{bmatrix} \mathbf{M}_{11}(\theta_2) & \mathbf{M}_{12}(\theta_2) \\ \mathbf{M}_{12}^T(\theta_2) & \mathbf{M}_{22}(\theta_2) \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \ddot{\mathbf{q}} \end{bmatrix} + \begin{bmatrix} \mathbf{F}_{\theta}(\theta, \mathbf{q}, \dot{\theta}, \dot{\mathbf{q}}) \\ \mathbf{F}_{\mathbf{q}}(\theta, \mathbf{q}, \dot{\theta}, \dot{\mathbf{q}}) \end{bmatrix} = \begin{bmatrix} \mathbf{T} \\ 0 \end{bmatrix} \quad (7.2)$$

with

$$\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} \qquad \mathbf{q} = [q_{1,1} \ldots q_{1,n_1} q_{2,1} \ldots q_{2,n_2}]^T \qquad \mathbf{T} = \begin{bmatrix} T_1 \\ T_2 \end{bmatrix}$$

$$\mathbf{F}_{\theta} = \begin{bmatrix} F_{\theta_1} \\ F_{\theta_2} \end{bmatrix} \qquad \mathbf{F}_{\mathbf{q}} = [F_{q_{1,1}} \ldots F_{q_{1,n_1}} F_{q_{2,1}} \ldots F_{q_{2,n_2}}]^T$$

$$\mathbf{M}_{11} = \begin{bmatrix} A_1 + 2A_c cos\theta_2 & I_{t2} + A_c cos\theta_2 \\ I_{t2} + A_c cos\theta_2 & I_{t2} \end{bmatrix}$$

$$\mathbf{M}_{12} = \begin{bmatrix} \mathbf{A}_{13} & \mathbf{A}_{14} \\ (1 \times n_1) & (1 \times n_2) \\ \\ \mathbf{A}_{23} & \mathbf{A}_{24} \\ (1 \times n_1) & (1 \times n_2) \end{bmatrix}$$

$$\mathbf{M}_{22} = \begin{bmatrix} \mathbf{A}_{33} & \mathbf{A}_{34} \\ (n_1 \times n_1) & (n_1 \times n_2) \\ \\ \mathbf{A}_{34}^T & \mathbf{A}_{44} \\ (n_2 \times n_1) & (n_2 \times n_2) \end{bmatrix}$$

where

$$
\begin{aligned}
A_{13}^{1j} &= K_{1,j} + \Phi_{1,j}(l_1)(m_p + m_{h2} + m_2)l_1 + \frac{d\Phi_{1,j}}{dx_1}(l_1)I_{t2} \\
&\quad +(\gamma_{cj} + \frac{d\Phi_{1,j}}{dx_1}(l_1)A_c)cos\theta_2 \\
A_{14}^{1j} &= \gamma_{2,j} + \mu_{2,j}l_1 cos\theta_2 \\
A_{23}^{1j} &= I_{t2}\frac{d\Phi_{1,j}}{dx_1}(l_1) + \gamma_{cj}cos\theta_2 \\
A_{24}^{1j} &= \gamma_{2,j} \\
A_{33}^{ij} &= m_{1,i,j} + \Phi_{1,i}(l_1)\Phi_{1,j}(l_1)(m_p + m_{h2} + m_2) \\
&\quad +\frac{d\Phi_{1,i}}{dx_1}(l_1)\frac{d\Phi_{1,j}}{dx_1}(l_1)I_{t2} \\
&\quad +(\gamma_{cj}\frac{d\Phi_{1,i}}{dx_1}(l_1) + \gamma_{ci}\frac{d\Phi_{1,j}}{dx_1}(l_1))\ cos\theta_2 \\
A_{34}^{ij} &= \frac{d\Phi_{1,i}}{dx_1}(l_1)\gamma_2 + \Phi_{1,i}(l_1)\mu_{2,j}cos\theta_2 \\
A_{44}^{ij} &= m_{2,i,j} + I_p\frac{d\Phi_{2,i}}{dx_2}(l_2)\frac{d\Phi_{2,j}}{dx_2}(l_2) + m_p\Phi_{2,i}(l_2)\Phi_{2,j}(l_2)
\end{aligned}
$$

where i : row index
j : column index

$$
\begin{aligned}
F_{\theta_1} &= -2A_c sin\theta_2\ \dot{\theta}_1\dot{\theta}_2 - A_c sin\theta_2\ \dot{\theta}_2^2 \\
&\quad + \sum_{j=1}^{n_1} 2(\gamma_{cj} - A_c\frac{d\Phi_{1,j}}{dx_1}(l_1))sin\theta_2\ \dot{\theta}_1\dot{q}_{1,j} \\
&\quad - \sum_{k=1}^{n_2} 2\mu_{2,k}l_1 sin\theta_2\ \dot{\theta}_1\dot{q}_{2,k} \\
&\quad -2\sum_{j=1}^{n_1} A_c\frac{d\Phi_{1,j}}{dx_1}(l_1)\ \ sin\theta_2\ \ \dot{\theta}_2\dot{q}_{1,j} \\
&\quad - \sum_{k=1}^{n_2} 2\mu_{2,k}l_1 sin\theta_2\ \dot{\theta}_2\dot{q}_{2,k} \\
&\quad - \sum_{j=1}^{n_1}\sum_{i=1}^{n_1} \gamma_{cj}\frac{d\Phi_{1,j}}{dx_1}(l_1)sin\theta_2\ \dot{q}_{1,i}\dot{q}_{1,j}
\end{aligned}
$$

$$
\begin{aligned}
&-2\sum_{j=1}^{n_1}\sum_{k=1}^{n_2}\mu_{2,k}\frac{d\Phi_{1,j}}{dx_1}(l_1)l_1 sin\theta_2\ \dot{q}_{2,k}\dot{q}_{1,j}\\
&-g\ sin\theta_1\sum_{j=1}^{n_1}[\eta_{1,j}+(m_2+m_{h2}+m_p)\Phi_{1,j}(l_1)]\ q_{1,j}\\
&-g\ sin(\theta_1+\theta_2)\sum_{k=1}^{n_2}\mu_{2,k}\ q_{2,k}\\
&+g\ (m_2+m_{h2}+m_p+\frac{m_1}{2})l_1 cos\theta_1\\
&+g\ (m_p+\frac{m_2}{2})l_2 cos(\theta_1+\theta_2)
\end{aligned}
\tag{7.3}
$$

$$
\begin{aligned}
F_{\theta_2} \;=\;& A_c sin\theta_2\ \dot{\theta}_1^2+\sum_{j=1}^{n_1}2\gamma_{cj}sin\theta_2\ \dot{q}_{1,j}\dot{\theta}_1\\
&+g\ [(m_p+\frac{m_2}{2})l_2 cos(\theta_1+\theta_2)-\sum_{k=1}^{n_2}\mu_{2,k}sin(\theta_1+\theta_2)\ q_{2,k}]
\end{aligned}
\tag{7.4}
$$

for j = 1,...,n_1:

$$
\begin{aligned}
F_{q_{1,j}} \;=\;& -[\gamma_{cj}-A_c\frac{d\Phi_{1,j}}{dx_1}(l_1)]sin\theta_2\ \dot{\theta}_1^2-2\gamma_{cj}sin\theta_2\ \dot{\theta}_1\dot{\theta}_2-\gamma_{cj}sin\theta_2\ \dot{\theta}_2^2\\
&+2\sum_{i=1}^{n_1}[\gamma_{ci}\frac{d\Phi_{1,j}}{dx_1}(l_1)-\gamma_{cj}\frac{d\Phi_{1,j}}{dx_1}(l_1)]sin\theta_2\ \dot{\theta}_1\dot{q}_{1,i}\\
&-\sum_{k=1}^{n_2}2\mu_{2,k}\Phi_{1,j}(l_1)sin\theta_2\ \dot{\theta}_1\dot{q}_{2,k}\\
&-2\sum_{i=1}^{n_1}\gamma_{cj}\frac{d\Phi_{1,i}}{dx_1}(l_1)sin\theta_2\ \dot{\theta}_2\dot{q}_{1,i}\\
&-\sum_{k=1}^{n_2}2\mu_{2,k}\Phi_{1,j}(l_1)sin\theta_2\ \dot{\theta}_2\dot{q}_{2,k}\\
&+\sum_{i=1}^{n_1}\sum_{k=1}^{n_2}\mu_{2,k}[\Phi_{1,i}(l_1)\frac{d\Phi_{1,j}}{dx_1}(l_1)\Phi_{1,j}(l_1)\frac{d\Phi_{1,i}}{dx_1}(l_1)]sin\theta_2\ \dot{q}_{1,i}\dot{q}_{2,k}\\
&+m'_{1,j}\omega_{1,j}^2 q_{1,j}+2m'_{1,j}\omega_{1,j}\zeta_{1,j}\dot{q}_{1,j}
\end{aligned}
$$

$$+g\ cos\theta_1[\eta_{1,j} + (m_2 + m_{h2} + m_p)\Phi_{1,j}(l_1)]$$
$$+g\ (m_p + \frac{m_2}{2})l_2\frac{d\Phi_{1,j}}{dx_1}(l_1)cos(\theta_1 + \theta_2) \qquad (7.5)$$

for k = 1,...,n_2:

$$F_{q_{2,k}} = \mu_{2,k}l_1 sin\theta_2\ \dot{\theta}_1^2 + \sum_{j=1}^{n_1}\mu_{2,k}[2\Phi_{1,j}(l_1) + l_1\frac{d\Phi_{1,j}}{dx_1}(l_1)]\ sin\theta_2\ \dot{\theta}_1\dot{q}_{1,j}$$
$$+m'_{2,k}\omega_{2,k}^2\ q_{2,k} + 2m'_{2,k}\omega_{2,k}\zeta_{2,k}\ \dot{q}_{2,k}$$
$$+g\mu_{2,k}cos(\theta_1 + \theta_2) \qquad (7.6)$$

In this model the following approximations are made:

- the dependency of the generalized mass matrix on flexible coordinates is neglected.
- in the nonlinear functions second order terms (terms with products of two flexible coordinates and derivatives of rigid coordinates (for example $q_{2,k}^2\dot{\theta}_2$ or $\dot{q}_{1,j}q_{2,k}\dot{\theta}_1$) , terms with products of more than two flexible coordinates (for example $q_{1,j}\dot{q}_{2,k}^2$), etc.) are neglected.
- in the gravity terms $cos(\theta_1 + \theta_2 + \sum_{j=1}^{n_1}\frac{d\Phi_{1,j}}{dx_1}(l_1)q_j)$ is approximated by $cos(\theta_1 + \theta_2)$

The mode shapes and eigenpulsations of this assumed modes model are obtained from the analytical solution of an Euler–Bernouilli beam eigenfunction analysis:

$$\Phi_{1,j}(x_1) = cosh(\lambda_{1,j}x_1) - cos(\lambda_{1,j}x_1)$$
$$+\gamma_{1,j}\,[sin(\lambda_{1,j}x_1) - sinh(\lambda_{1,j}x_1)]$$

$$\Phi_{1,j}(x_2) = cosh(\lambda_{2,j}x_2) - cos(\lambda_{2,j}x_2)$$
$$+\gamma_{2,j}\,[sin(\lambda_{2,j}x_2) - sinh(\lambda_{2,j}x_2)]$$

$$\omega_{i,j} = \lambda_{i,j}^2\sqrt{\frac{(EI)_i}{\rho_i S_i}}$$

with $\gamma_{i,j}$ and $\lambda_{i,j}$ determined from the boundary conditions of the beam [40].

The numerical simulation model only retains one flexible mode per link so that the state variable x is composed of

- θ_1, θ_2 : rigid body angular displacement at the first and second hubs.
- $q_{1,1}$, $q_{2,1}$: first modal coordinates of the first and second links
- and their respective derivatives.

The input vector is composed of

- $[T_1 \quad T_2]^T$: torques applied at the first and second joints

The numerical values of the parameters of the simulation model are given in table 7.2.

7.3 Control Design

The generalized nonlinear decoupling controller is designed first as a feedforward decoupling controller, and second as a feedback decoupling controller (section 4.5.3).

The feedback gain K is determined from the linearized model given by equation 4.25 and 4.26 where

$$\begin{aligned} \mathbf{H_{q_0}} &= w_{11}\frac{\delta \mathbf{F_q}}{\delta \theta} \\ &= w_{11}\begin{bmatrix} \frac{\delta F_{q_{1,1}}}{\delta \theta_1} & \frac{\delta F_{q_{1,1}}}{\delta \theta_2} \\ \\ \frac{\delta F_{q_{2,1}}}{\delta \theta_1} & \frac{\delta F_{q_{2,1}}}{\delta \theta_2} \end{bmatrix} \end{aligned}$$

with

$$\begin{aligned} \frac{\delta F_{q_{1,1}}}{\delta \theta_1} &= -g\ sin\theta_{1d}[\eta_{1,1} + (m_2 + m_{h2} + m_p)\Phi_{1,1}(l_1)] \\ &\quad -g\ (m_p + \frac{m_2}{2})l_2\frac{d\Phi_{1,1}}{dx_1}(l_1)sin(\theta_{1d} + \theta_{2d}) \\ \frac{\delta F_{q_{1,1}}}{\delta \theta_2} &= -g\ (m_p + \frac{m_2}{2})l_2\frac{d\Phi_{1,1}}{dx_1}(l_1)sin(\theta_{1d} + \theta_{2d}) \\ \frac{\delta F_{q_{2,1}}}{\delta \theta_1} &= \frac{\delta F_{q_{2,1}}}{\delta \theta_2} = -g\mu_{2,k}sin(\theta_{1d} + \theta_{2d}) \end{aligned} \tag{7.7}$$

First joint:	
inertia around Z	
through O_1	$I_{h1} = 3.5\ kgm^2$
First flexible beam:	
length	$l_1 = 0.5\ m$
cross section	$S_1 = 0.00032\ m^2$
geometric moment	
of inertia	$I_1 = 0.17110^{-8}\ m^4$
Young modulus	$E_1 = 0.7\ 10^{11}\ N/m^2$
density	$\rho_1 = 2800\ kg/m^3$
mass	$m_1 = 0.448\ kg$
inertia around Z	
through O_1	$I_{b1} = 0.0373\ kgm^2$
Second joint:	
mass	$m_{h2} = 1\ kg$
inertia around Z	
through O_2	$I_{h2} = 0.79\ kgm^2$
Second flexible beam = First flexible beam	
Payload:	
mass	$m_p = 3\ kg$
inertia around Z	
through G	$I_p = 0.00244\ kgm^2$
One mode is retained per link; the damping ratio is equal to 1 % for each mode.	

Table 7.2: Numerical parameters of the simulation model.

$$
\begin{aligned}
\mathbf{H}_{\mathbf{q}_f} &= w_{11}\frac{\delta \mathbf{F_q}}{\delta \mathbf{q}} \\
&= w_{11}\begin{bmatrix} m'_{1,1}\omega^2_{1,1} & 0 \\ 0 & m'_{2,1}\omega^2_{2,1} \end{bmatrix} \\
\mathbf{H}_{\dot{\mathbf{q}}_0} &= w_{11}\frac{\delta \mathbf{F_q}}{\delta \dot{\theta}} \\
&= w_{11}\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}
\end{aligned}
$$

$$\begin{aligned} \mathbf{H}_{\dot{\mathbf{q}}_f} &= w_{11}\frac{\delta \mathbf{F}_\mathbf{q}}{\delta \dot{\mathbf{q}}} \\ &= w_{11}\begin{bmatrix} 2m'_{1,1}\omega_{1,1}\zeta_{1,1} & 0 \\ 0 & 2m'_{2,1}\omega_{2,1}\zeta_{2,1} \end{bmatrix} \end{aligned} \tag{7.8}$$

and $w_{11} = (\mathbf{W}_{12}^T\mathbf{W}_{11}^{-1}\mathbf{W}_{12} - \mathbf{W}_{22})(\theta_{2d})$.

The feedback gains are determined with a pole placement algorithm, based on the continuous time model. This is allowed because of the high sample frequency of 500 Hz used in the simulations. The poles are placed at

$$\begin{aligned} &-2\pi(3.64 \pm 0.48\, i) \\ &-2\pi\ 6.15 \\ &-2\pi(6.82 \pm 4.33\, i) \\ &-2\pi(7.52 \pm 10.39\, i) \\ &-2\pi\ 10.73 \end{aligned} \tag{7.9}$$

As a result, the bandwidth of the closed loop system is approximately 3.7 Hz, while the lowest resonance frequency of the links is 4 Hz.

The desired state variables, θ_{1d}, θ_{2d}, $q_{1,1d}$, $q_{2,1d}$ and derivatives, are calculated using the procedure described in section 5.4.1. They are calculated from the equations describing the relationship between the virtual angular link positions θ_{1r} and θ_{2r} and the reference values of the state variables (equation 5.10).

$$\begin{aligned} \theta_{1d} &= \theta_{r1} - \frac{\Phi_{1,1}(l_1)}{l_1}q_{1,1d} \\ \theta_{2d} &= \theta_{r2} + \frac{\Phi_{1,1}(l_1)}{l_1}q_{1,1d} - \frac{\Phi_{2,1}(l_2)}{l_2}q_{2,1d} - \frac{d\Phi_{1,1}}{dx_1}(l_1)q_{1,1d} \end{aligned} \tag{7.10}$$

and the equations describing the dynamic behaviour of the flexible coordinates

$$\mathbf{M}_{12}^T(\theta_2)\begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \mathbf{M}_{22}(\theta_2)\begin{bmatrix} \ddot{q}_{1,1} \\ \ddot{q}_{2,1} \end{bmatrix} + \begin{bmatrix} F_{q_{1,1}}(\theta, \mathbf{q}, \dot{\theta}, \dot{\mathbf{q}}) \\ F_{q_{2,1}}(\theta, \mathbf{q}, \dot{\theta}, \dot{\mathbf{q}}) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{7.11}$$

To simplify the calculations and to obtain a numerically stable solution equation 7.11 is simplified to

$$\mathrm{M}_{12}^{T}(\theta_2)\begin{bmatrix}\ddot{\theta}_1\\ \ddot{\theta}_2\end{bmatrix}+\begin{bmatrix}F'_{q_{1,1}}(\theta,\dot{\theta},q_{1,1})\\ F'_{q_{2,1}}(\theta,\dot{\theta},q_{2,1})\end{bmatrix}=\begin{bmatrix}0\\0\end{bmatrix} \tag{7.12}$$

with

$$\begin{aligned}
F'_{q_{1,1}} &= -[\gamma_{c1}-A_c\frac{d\Phi_{1,1}}{dx_1}(l_1)]sin\theta_2\ \dot{\theta}_1^2-2\gamma_{c1}sin\theta_2\ \dot{\theta}_1\dot{\theta}_2\\
&\quad -\gamma_{c1}sin\theta_2\ \dot{\theta}_2^2+m'_{1,1}\omega_{1,1}^2q_{1,1}\\
&\quad +g\ cos\theta_1[\eta_{1,1}+(m_2+m_{h2}+m_p)\Phi_{1,1}(l_1)]\\
&\quad +g\ (m_p+\frac{m_2}{2})l_2\frac{d\Phi_{1,1}}{dx_1}(l_1)cos(\theta_1+\theta_2)\\
F'_{q_{2,1}} &= \mu_{2,1}l_1sin\theta_2\ \dot{\theta}_1^2+m'_{2,1}\omega_{2,1}^2\ q_{2,1}\\
&\quad +g\mu_{2,1}cos(\theta_1+\theta_2)
\end{aligned} \tag{7.13}$$

$q_{1,1d}$ and $q_{2,1d}$ are then calculated as

$$\begin{aligned}
q_{1,1d} &= \frac{1}{m'_{1,1}\omega_{1,1}^2}\{-\mathrm{M}_{12}^{T}(1,1)\ddot{\theta}_{1d}-\mathrm{M}_{12}^{T}(1,2)\ddot{\theta}_{2d}\\
&\quad +[\gamma_{c1}-A_c\frac{d\Phi_{1,1}}{dx_1}(l_1)]sin\theta_{2d}\ \dot{\theta}_{1d}^2+2\gamma_{c1}sin\theta_{2d}\ \dot{\theta}_{1d}\dot{\theta}_{2d}\\
&\quad +\gamma_{c1}sin\theta_{2d}\ \dot{\theta}_{2d}^2\\
&\quad -g\ cos\theta_{1d}[\eta_{1,1}+(m_2+m_{h2}+m_p)\Phi_{1,1}(l_1)]\\
&\quad -g\ (m_p+\frac{m_2}{2})l_2\frac{d\Phi_{1,1}}{dx_1}(l_1)cos(\theta_{1d}+\theta_{2d})\}\\
q_{2,1d} &= \frac{1}{m'_{2,1}\omega_{2,1}^2}\{-\mathrm{M}_{12}^{T}(2,1)\ddot{\theta}_{1d}-\mathrm{M}_{12}^{T}(2,2)\ddot{\theta}_{2d}\\
&\quad -\mu_{2,1}l_1sin\theta_{2d}\ \dot{\theta}_{1d}^2-g\mu_{2,1}cos(\theta_{1d}+\theta_{2d})\}
\end{aligned} \tag{7.14}$$

where in a first step θ_{1d},θ_{2d}, $\dot{\theta}_{1d}$,$\dot{\theta}_{2d}$, $\ddot{\theta}_{1d}$ and $\ddot{\theta}_{2d}$ are taken equal to the virtual positions, velocities and accelerations.

The corrected trajectories for the joint angles, θ_{1d} and θ_{2d}, are calculated from equation 7.10. The derivatives are calculated by differentiation.

To obtain a more accurate result, the calculation of $q_{1,1d}$ and $q_{2,1d}$ from equation 7.14 is iterated, taking the corrected joint angles in

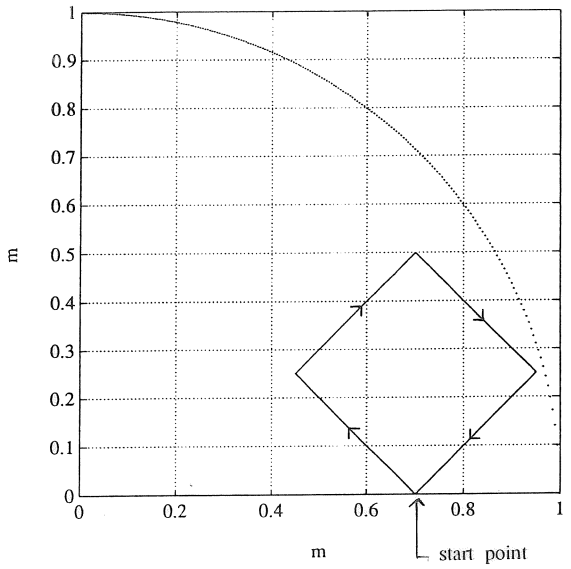

Figure 7.2: Solid line: desired trajectory in the x–y plane. Dashed line: working space of the robot.

the righthand side of the equations, as obtained from equation 7.10 (to avoid differentiation errors, the virtual velocities and accelerations are still used). This still gives a small correction of $q_{1,1d}$ and $q_{2,1d}$. Convergence is observed after one or two steps.

The deflection due to gravity (the last terms in equation 7.14) is called static deflection, the deflection due to inertial, centrifugal and coriolis forces (the other terms in equation 7.14) is called quasi–static deflection.

7.4 Simulation results

The generalized nonlinear decoupling method is applied to the simulation model. Figure 7.2 shows the desired trajectory in the x–y plane, to be followed by the two flexible link robot. The desired trajectory has the form of a square. The robot has to stop in each corner point of the square. Each side has to be travelled in 1 second. Motion takes place in a vertical plane. The trajectory is generated based on a ninth order polynomial (appendix C). The use of such high order polynomials limits the excitation of the structure.

First a rigid nonlinear controller, as explained in section 4.3, is simulated. In order to compare with the generalized nonlinear decoupling controller, the static deflection due to gravity is compensated in the calculation of the desired motor positions. If the bandwidth of this controller is sufficiently lower than the first resonance frequency, this control method gives good results in terms of overshoots and oscillations. Because the dynamic deflection is not taken into account, the tracking accuracy is limited. However if the bandwidth of the controller reaches the resonance frequency, the closed loop dynamic behaviour deteriorates very rapidly. The structure is severely excited and the resulting vibrations of the tips of the flexible links are damped out rather slowly. This can be seen in figure 7.3, which gives the flexible displacement of the tip of link 1 and link 2, $v_1(l_1) = \Phi_{1,1}(l_1)q_{1,1}$, and $v_2(l_2) = \Phi_{2,1}(l_2)q_{2,1}$, and the simulated and desired motor positions. The results for the first side of the square are shown. Here the bandwidth of the controller is 3.7 Hz, while the lowest resonance frequency of the links is 4 Hz.

Then the generalized nonlinear decoupling controller is simulated. The feedback gains are calculated with respect to the final desired position of the first side of the square. Stable behaviour is obtained by using these feedback gains for the whole trajectory. As in the previous case the bandwidth of the controller is chosen approximately equal to the first resonance frequency of the open loop system (3.7 Hz). The desired joint angles, velocities and accelerations, and the desired flexible coordinates and their derivatives in function of time are calculated in three different ways:

1. Neglecting of the flexibility of the robot: the desired joint angles are the joint angles of the virtual rigid links.

2. Compensation of the static deflection due to gravitation: the desired joint angles and flexible coordinates are calculated considering only the static behaviour of the robot. This is obtained by putting the first and second derivatives of joint angles and flexible coordinates equal to zero in the model.

3. Compensation of static and dynamic deflection (quasi–static deflection) : the desired joint angles and flexible coordinates are calculated based on the total dynamic model as described in the previous section.

Figures 7.4, 7.5 and 7.6 show the tip position and the tracking error normal and tangential to the desired trajectory for the three controllers. (The total tracking error is projected along directions normal and tangential to the desired trajectory.) If the static and dynamic deflection is compensated, the results are nearly perfect. The maximum tracking error normal to the desired trajectory is 0.7 mm, the maximum tracking error tangential to the desired trajectory is 0.6 mm. If only the static deflection is compensated the tracking errors are much higher: the maximum tracking error normal to the desired trajectory is 28 mm, the maximum tracking error tangential to the desired trajectory is 21 mm. Without compensation of flexibility, in addition to the dynamic tracking errors, static errors are observed.

Figure 7.7 shows the simulated and desired motor positions and flexible displacements of the endpoints of the links for the generalized nonlinear decoupling control with compensation of static deflection in the desired state calculation (controller 2). In order to show that vibrations are damped out very rapidly, figure 7.8 shows the results for the first side of the square. The robot has to stop in the second corner point of the square. Comparison with the rigid controller (figure 7.3) shows the improved dynamic behaviour.

Figure 7.9 shows the simulated and desired motor positions and flexible displacements of the endpoints of the links for the generalized nonlinear decoupling control with compensation of static and dynamic deflection in the desired state calculation (controller 3). It can be concluded that the approximations made in the calculation of desired flexible coordinates are justified: the desired state variables are approximately equal to the simulated state variables. Also the total motor torques and the nonlinear decoupling torque (motor torque without linear feedback of state variables) are shown. The nonlinear decoupling torque provides the main part of the control action, the additional linear feedback of the full state $\mathbf{x}$ is only needed to correct for small deviations in the trajectory tracking and to ensure proper stability of the system about the desired state $\mathbf{x}_d$.

The figures shown give the simulation results obtained by applying the nonlinear feedback control scheme. By applying the feedforward scheme nearly identical results are obtained.

7.5 Conclusion

The generalized nonlinear decoupling controller is applied to a simulation model of a two link flexible robot. The simulations have shown that in comparison with a classical rigid controller the vibrations are damped out very rapidly. They have also shown the importance of the desired state calculation. If the desired joint angles and flexible coordinates are calculated based on the total dynamic model, compensation of static and dynamic deflection is obtained and the tracking errors are reduced to negligible values.

In this chapter a perfect dynamic model is supposed and both the simulation model and controller model considers only the first flexible mode. The sensitivity to model accuracy of the generalized nonlinear decoupling controller and the spill–over problem is not dealt with. Simple simulations with "perturbed" models (in the control model some parameters have been changed with maximum 50 %) have shown no problem of stability, however the tracking error increases considerably. The experiments (chapter 6,8 and 9) show that model errors, spill–over and also unaccurate modeling of nonlinear motor friction limits the achievable bandwidth of the controller.

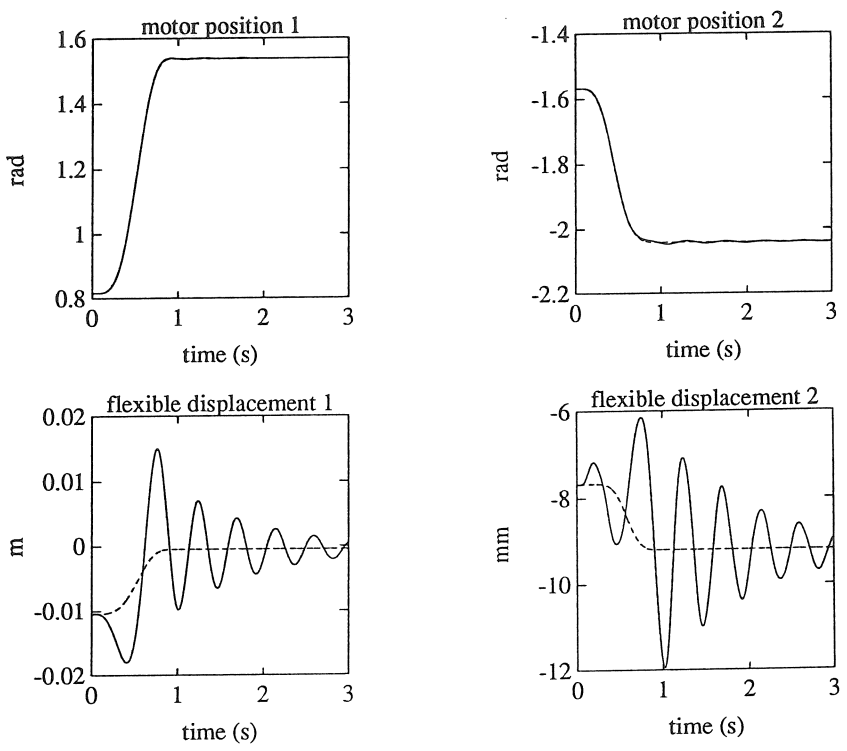

Figure 7.3: Classical rigid body control. Upper figures: simulated (solid line) and desired (dashed line) motor positions. Bottom figures: simulated (solid line) and desired (dashed line) flexible displacements of the endpoints of the links.

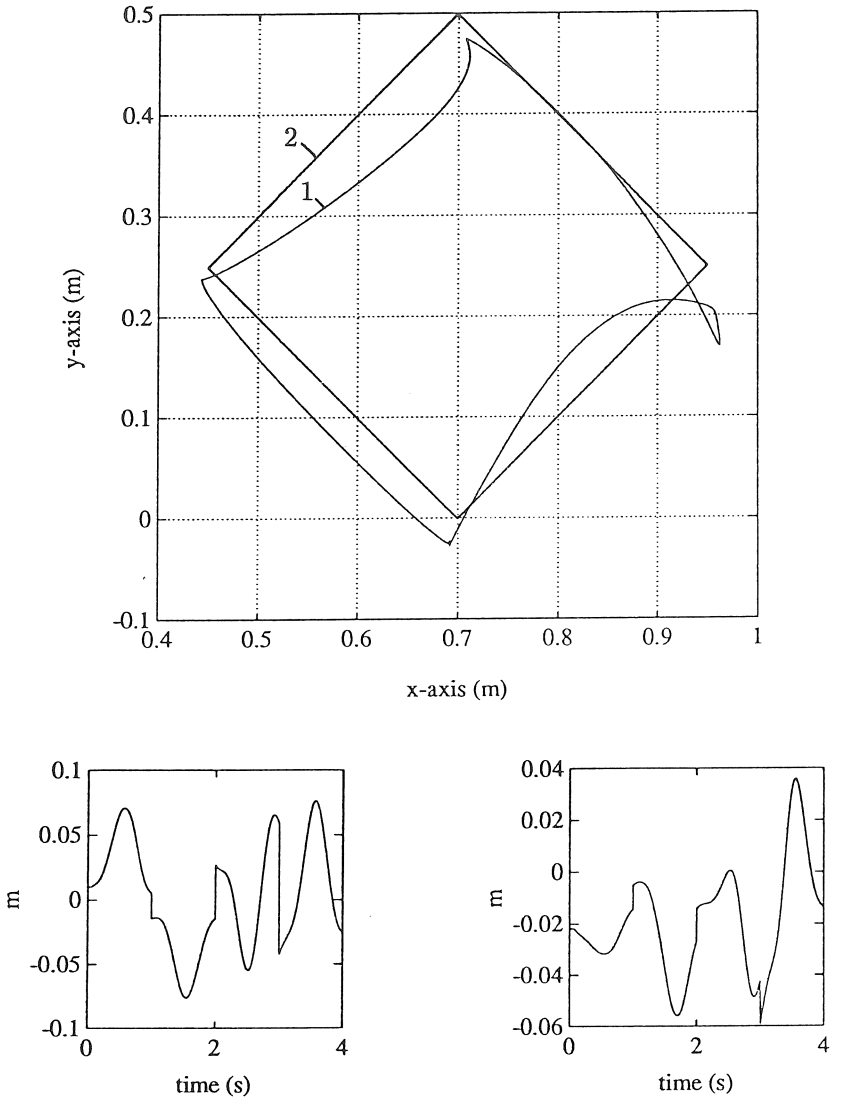

Figure 7.4: Generalized nonlinear decoupling control without compensation of flexibility (controller 1). Upper figure: simulated (1) and desired (2) tip position. Bottom figures: tracking error normal (left figure) and tangential (right figure) to the desired trajectory in function of time. (The total tracking error is projected along directions normal and tangential to the desired trajectory.)

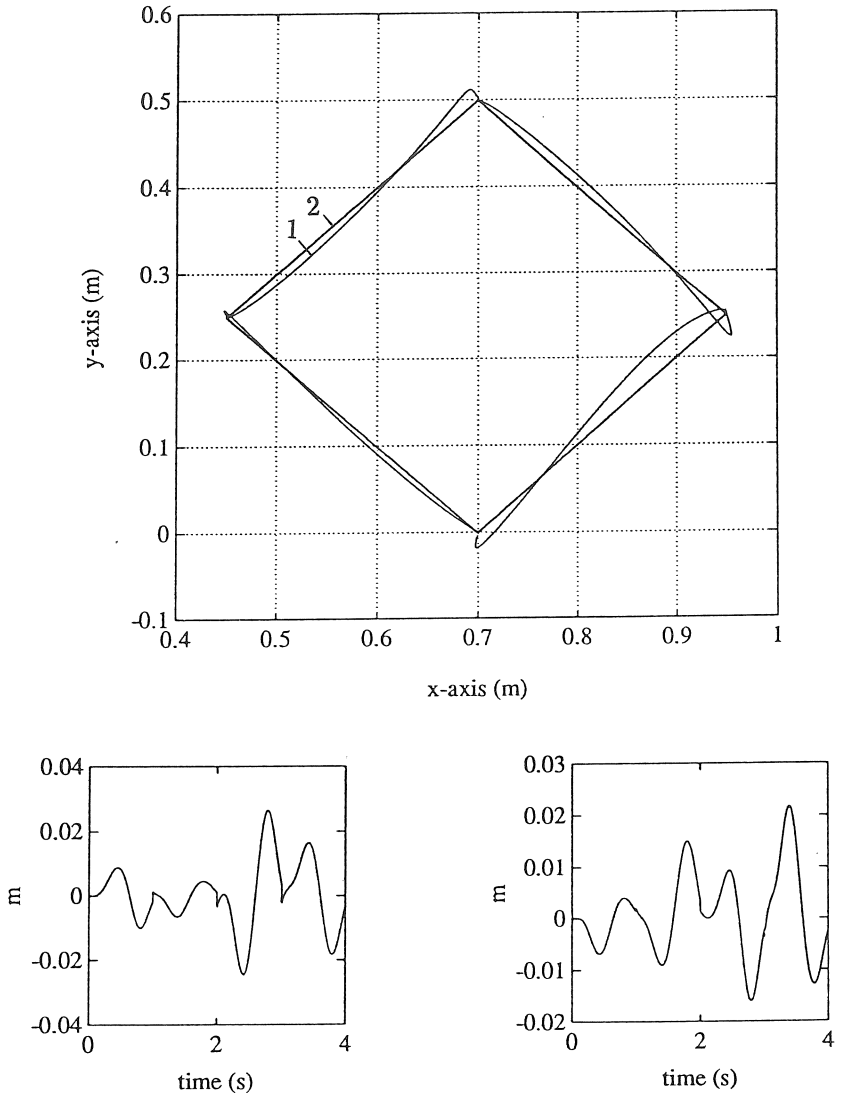

Figure 7.5: Generalized nonlinear decoupling control with compensation of static deflection (controller 2). Upper figure: simulated (1) and desired (2) tip position. Bottom figures: tracking error normal (left figure) and tangential (right figure) to the desired trajectory in function of time.

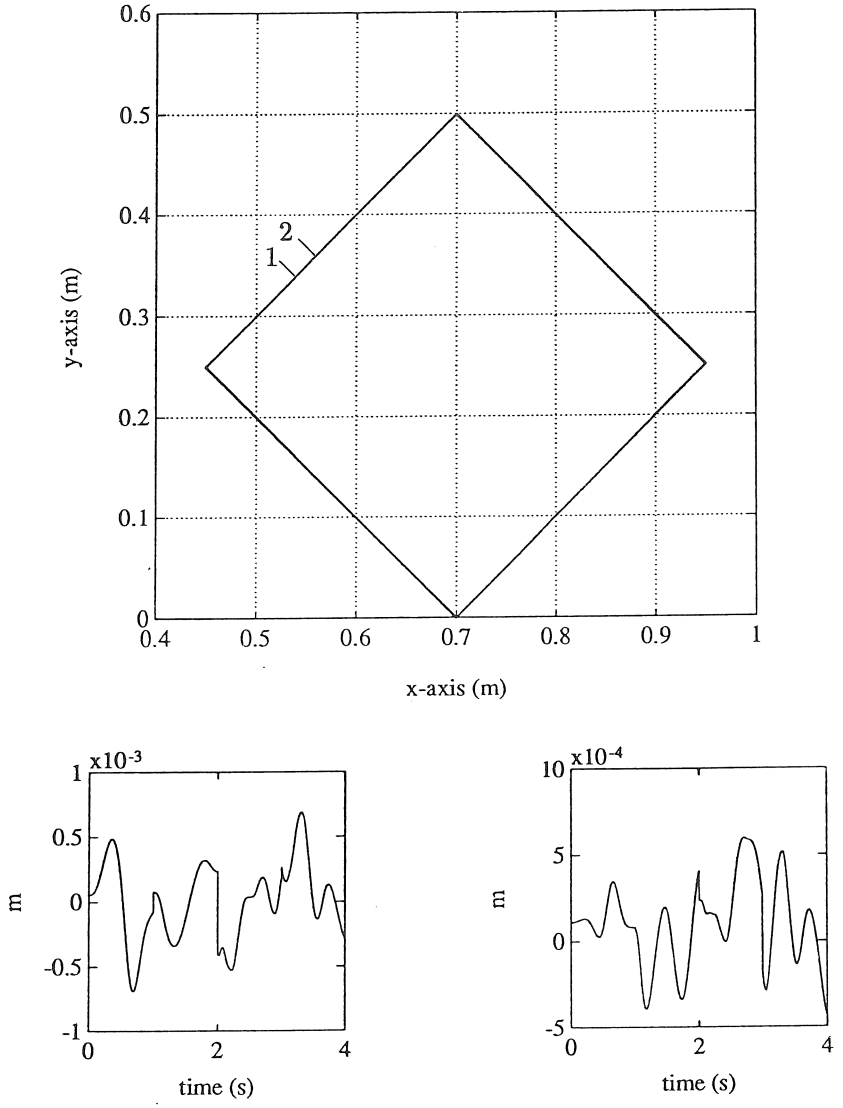

Figure 7.6: Generalized nonlinear decoupling control with compensation of static and dynamic deflection (controller 3). Upper figure: simulated (1) and desired (2) tip position. Bottom figures: tracking error normal (left figure) and tangential (right figure) to the desired trajectory in function of time.

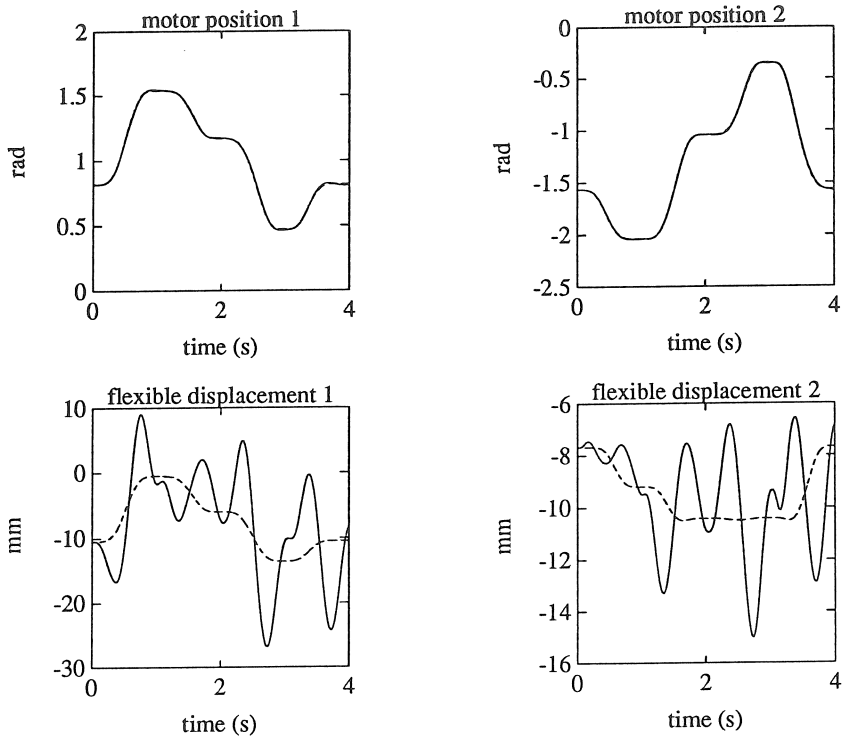

Figure 7.7: Generalized nonlinear decoupling control with compensation of static deflection (controller 2). Upper figures: simulated (solid line) and desired (dashed line) motor positions. Bottom figures: simulated (solid line) and desired (dashed line) flexible displacements of the endpoints of the links.

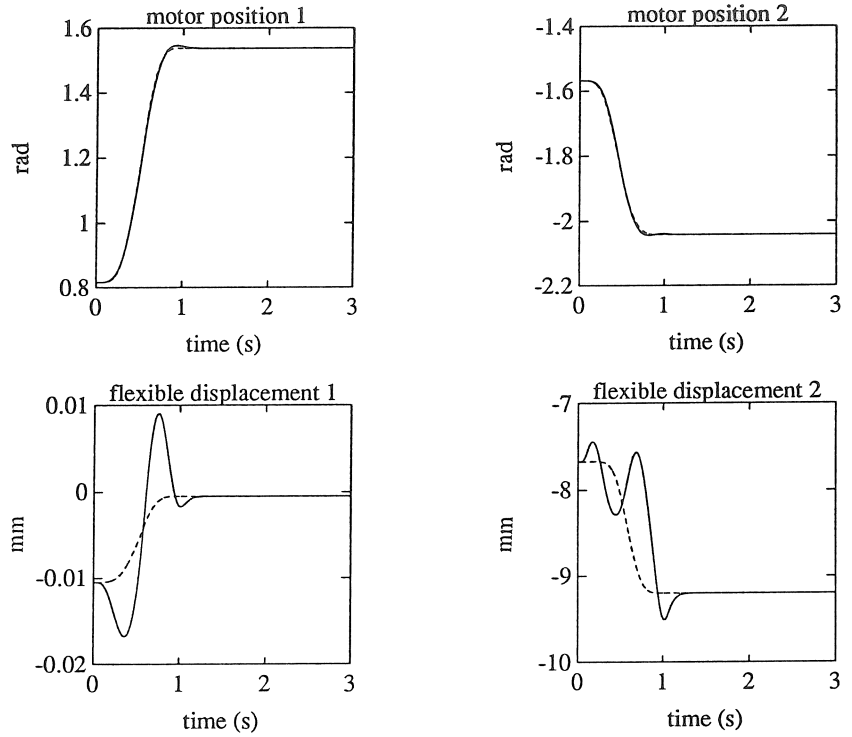

Figure 7.8: Generalized nonlinear decoupling control with compensation of static deflection (controller 2): results for the first side of the square. Upper figures: simulated (solid line) and desired (dashed line) motor positions. Bottom figures: simulated (solid line) and desired (dashed line) flexible displacements of the endpoints of the links.

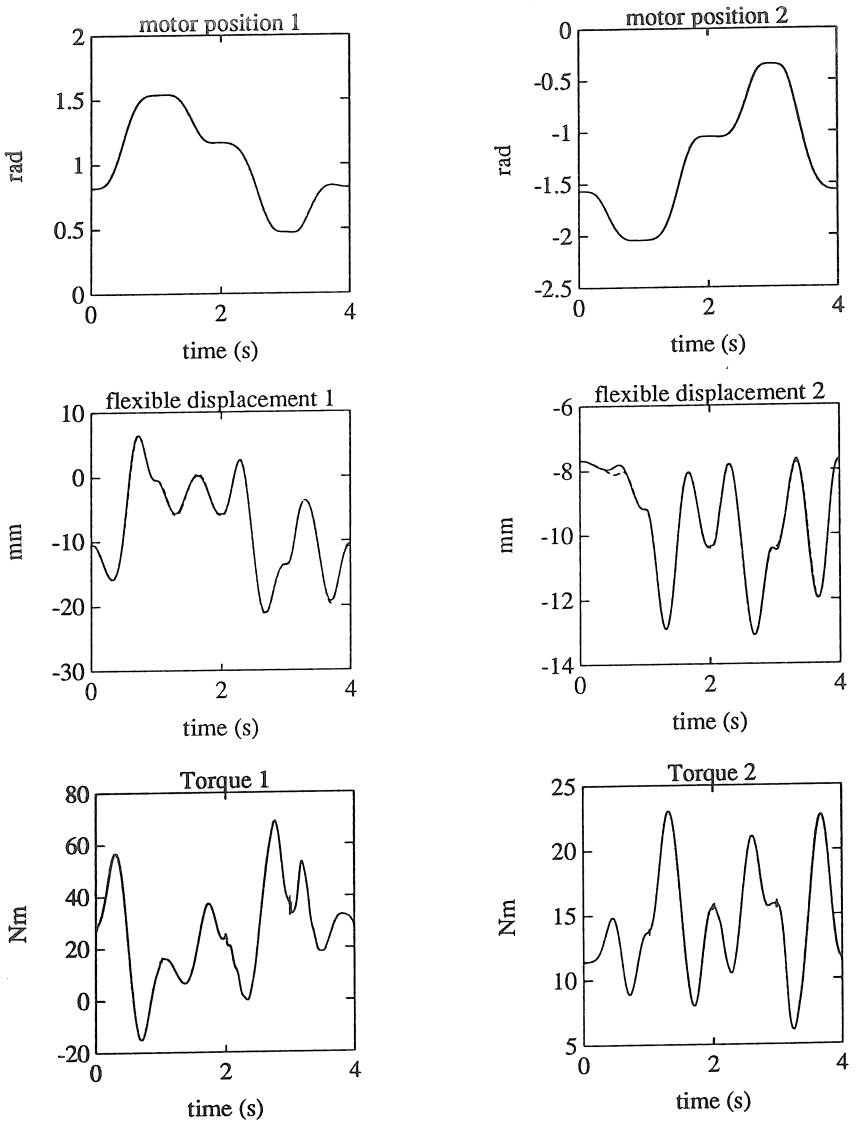

Figure 7.9: Generalized nonlinear decoupling control with compensation of static and dynamic deflection (controller 3). Upper figures: simulated (solid line) and desired (dashed line) motor positions. Middle figures: simulated (solid line) and desired (dashed line) flexible displacements of the endpoints of the links. Bottom figures: total torques (solid line) and nonlinear decoupling torques (dashed line).

Chapter 8

Experimental results on a two–link robot with a flexible first joint

8.1 Introduction

This chapter reports on experimental experience with identification and control of a two–link robot with a flexible first joint.

The proposed nonlinear control method for flexible robot structures is implemented on a test structure which consists of a series combination of two beams driven by direct drive motors. The first beam is connected by springs to the motor. This simulates a flexible first joint. A nonlinear model describing the dynamic behaviour of the system is identified. The relatively high nonlinear Coulomb friction in the system is taken into account in the identification. This results in more reliable estimates for the model parameters. The friction is also compensated in the control. To estimate the velocities a nonlinear discrete state estimator is used. And the trajectory calculation algorithm takes the flexibility of the structure into account.

Experimental results show the superior performance of this controller compared to classical rigid control. The tracking error is reduced by a factor 10 and the steady state error by a factor 35.

8.2 Description of the test set up and mathematical formulation of the model.

Figure 8.1 gives a schematic representation of the test set up. The real set up is shown in figure 8.2. It consists of a series combination of two

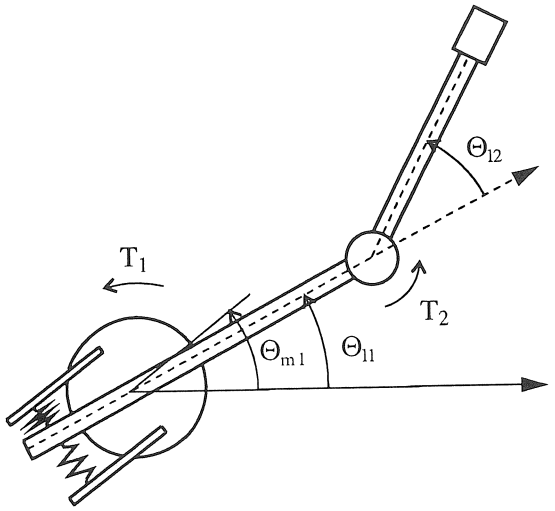

Figure 8.1: Schematic representation of the test set up

beams driven by direct drive motors. The length of the first beam is 0.499 m. The length of the second beam is 0.566 m. The motion takes place in the horizontal plane. The first beam is connected by springs to the motor. This simulates a flexible first joint. The first resonance frequency varies from $4.6Hz$ to $4.8Hz$ as a function of the position of the second link. The motor positions are measured by encoders. The encoder of the first motor has a resolution of 1024000 pulses per revolution. Hence, one encoder pulse corresponds to an angle of 6.1359 10^{-6} radians. The encoder of the second motor has a resolution of 655360 pulses per revolution. Hence, one encoder pulse corresponds to an angle of 9.5874 10^{-6} radians. The deformation of the spring is measured by a LVDT (linear variable differential transformer). The calibration constant of the LVDT, the bridge and the A/D convertor is 4.071 10^{-5} radians per bit of the A/D convertor. During the cali-

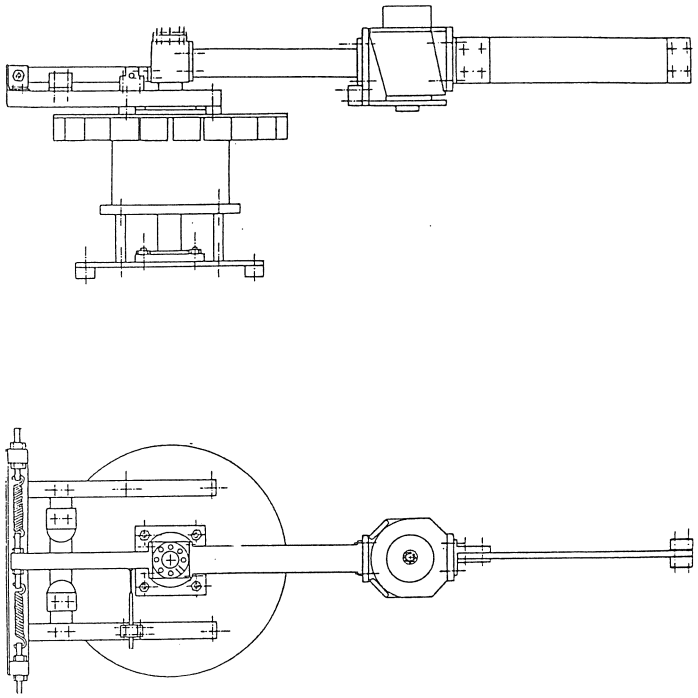

Figure 8.2: The real set up

bration experiment, the tip position of the first link is held constant, while the motor angle and the deformation of the spring are increased. The deformation of the spring is then equal to the motor displacement. The control input to the system is a voltage between -10 Volt and $+10$ Volt, which is converted by the power supply of the motor into a current. The motor torque is proportional to the current. The maximium motor torque of the first motor is 200 Nm. The maximium motor torque of the second motor is 45 Nm.

The total dynamic model is derived applying Lagrange's equations as described in chapter 2. The nonlinear equations of motion are given by:

$$I_{m1}\ddot{\theta}_{m1} + T_{f1} - K_{j1}(\theta_{l1} - \theta_{m1}) = T_1 \tag{8.1}$$

$$(A_2 + A_c\, cos(\theta_{l2}))\ \ddot{\theta}_{l1} + A_2\ \ddot{\theta}_{l2} + A_c\ \dot{\theta}_{l1}^2 sin(\theta_{l2}) + T_{f2} = T_2 \tag{8.2}$$

$$(A_1 + 2\ A_c\, cos(\theta_{l2}))\ \ddot{\theta}_{l1} + (A_2 + A_c\, cos(\theta_{l2}))\ \ddot{\theta}_{l2} - 2\ A_c\ \dot{\theta}_{l1}\dot{\theta}_{l2} sin(\theta_{l2})$$

$$- A_c \, \dot{\theta}_{l2}^2 sin(\theta_{l2}) + K_{j1}(\theta_{l1} - \theta_{m1}) = 0 \qquad (8.3)$$

where θ_{m1} : motor position 1
θ_{l1} : beam position 1
$\theta_{l1} - \theta_{m1} = q_{j1}$: deformation of the spring
θ_{l2} : beam position 2 = motor position 2
T_1 : motor torque 1
T_2 : motor torque 2
I_{m1} : rotor inertia 1
T_{f1} : motor friction torque 1
T_{f2} : motor friction torque 2
K_{j1} : spring constant
A_2 $= I_{m2} + I_{b2} + I_p + m_p l_2^2$: Total inertia 2
A_1 $= I_{b1} + I_{b2} + I_{m2} + I_p + m_{b2} l_1^2 + m_{m2} l_1^2 + m_p l_1^2 + m_p l_2^2$
A_c $= l_1 l_2 (\frac{m_{b2}}{2} + m_p)$

Initially, it is assumed that the motor frictions consist only of linear viscous friction:

$$T_{f1} = C_{m1} \, \dot{\theta}_{m1}$$

$$T_{f2} = C_{m2} \, \dot{\theta}_{l2}$$

In this model the cross–coupling terms due to motor mass are not neglected. Hence, the form of the equations is different from the form described in section 2.4. The reason is that direct drive motors are used here. The first assumption on page 27 (high gear transmission rates) is thus not valid.

8.3 Identification

The unknown parameters in this model have to be identified.

Because the dynamic model is nonlinear, the classical identification schemes based on a linear input-output model, as described in chapter 3, can not be used.

An identification approach is proposed which starts directly from the nonlinear equations of motion: A known input torque is applied to the system and the corresponding positions and deformation of

the spring are measured. Introducing the measured values and their first and second derivatives in the nonlinear equations gives rise to an overdetermined set of linear equations. For example, equation 8.1 becomes:

$$\begin{bmatrix} \ddot{\theta}_{m1}(i) & \dot{\theta}_{m1}(i) & -(\theta_{l1}(i) - \theta_{m1}(i)) \end{bmatrix} \begin{bmatrix} I_{m1} \\ C_{m1} \\ K_{j1} \end{bmatrix} = T_1(i) \quad i = 1, ..., N$$

with N the number of measurement samples and $\begin{bmatrix} I_{m1} & C_{m1} & K_{j1} \end{bmatrix}^T$ the unknown parameter vector. This set is solved in a linear least squares sense to obtain the unknown parameters of the model.

The applied input torque is a band limited random signal. The cut–off frequency of the input signal spectrum is 20 Hz. This band limited random signal excites the system in the frequency band of interest.

To obtain accurate identification results, prefiltering of input and output data is necessary. For this purpose a second order digital low pass Butterworth filter with a cut–of frequency at 10 Hz is used. The input–output signals are measured with a sample frequency, f_s, of 100 Hz. The same sample frequency is used for the control. The derivatives are calculated with a fifth order Gear's algorithm:

$$\dot{y}_k = [137y_k - 300y_{k-1} + 300y_{k-2} - 200y_{k-3} + 75y_{k-4} - 12y_{k-5}]\ f_s/60 \tag{8.4}$$

The practice has shown that trying to identify all the parameters in one step, gives unreliable identification results. The reason is the different order of magnitude of the terms in the equations. Therefore the identification process is divided into different steps.

- Step 1: Identification of the motor inertias and friction coefficients.
  The measurements on the direct drive motors are performed on two separate set ups. The dynamic behaviour is described by:

$$T_1 = I_{m1}\ddot{\theta}_{m1} + C_{m1}\dot{\theta}_{m1}$$

$$T_2 = I_{m2}\ddot{\theta}_{l2} + C_{m2}\dot{\theta}_{l2}$$

  It turns out that the Coulomb friction is not negligible. Therefore the total motor friction is modeled as:

$$T_{f1} = C_{m1}\dot{\theta}_{m1} + F_{m1}sgn(\dot{\theta}_{m1}) \qquad (motor1) \tag{8.5}$$

$$T_{f2} = C_{m2}\dot{\theta}_{l2} + F_{m2}sgn(\dot{\theta}_{l2}) \quad (motor2) \qquad (8.6)$$

where F_{m1} : Coulomb friction of motor 1
C_{m1} : linear viscous friction of motor 1
F_{m2} : Coulomb friction of motor 2
C_{m2} : linear viscous friction of motor 2
$sgn(v) = 1 \quad if\ v > 0$
$= 0 \quad if\ v = 0$
$= -1 \quad if\ v < 0$

The parameters C_{m1}, C_{m2}, F_{m1} and F_{m2} are identified with the following measurement procedure [37]: After bringing the motor to a certain velocity the torque is put to zero. While the velocity is decaying to zero the position is measured. The dynamic behaviour is described by:

$$-I_{m1}\,\ddot{\theta}_{m1} = C_{m1}\dot{\theta}_{m1} + F_{m1}\,sgn(\dot{\theta}_{m1}) \quad (motor1)$$
$$-I_{m2}\,\ddot{\theta}_{l2} = C_{m2}\dot{\theta}_{l2} + F_{m2}\,sgn(\dot{\theta}_{l2}) \quad (motor2)$$

In these equations the identified rotor inertias are supposed to be exactly known. Solving these equations in a least squares sense gives the values of the friction coefficients. The measured and identified relationship between velocity and friction torque are compared in figure 8.3.

- Step 2: Equation 8.1 is identified. This gives an estimate of the spring constant K_{j1}.
- Step 3: Equation 8.2 and 8.3 are solved together. The motor friction, and the already identified spring constant are considered as known parameters: this means that the terms in these parameters are put in the right hand side of the equations. This step gives an estimate of the parameters A_1, A_2, and A_c.

This identification process is repeated for different noise measurements, in different positions. Table 8.1 shows the coincidence between the identified and analytical values. The analytical values are given characteristics of motors and springs and are calculated from geometrical properties. In addition, table 8.1 gives the standard deviation on the identified parameters for different measurements.

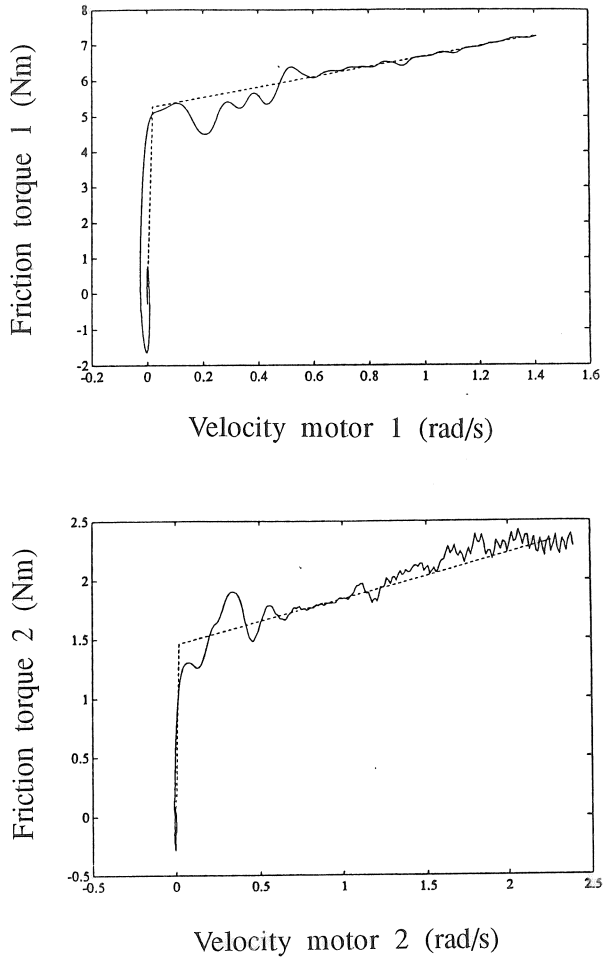

Figure 8.3: measured (solid line) and identified (dashed line) relationship between velocity and friction torque.

Analytical parameters		Identified parameters	deviation
I_{m1}	$1.6\ kgm^2$	$1.936\ kgm^2$	2 %
C_{m1}	–	$1.4\ kgm^2/s$	–
F_{m1}	–	$5.28\ Nm$	–
C_{m2}	–	$0.4\ kgm^2/s$	–
F_{m2}	–	$1.37\ Nm$	–
K_{j1}	$1150\ Nm/rad$	$1223\ Nm/rad$	1.3 %
A_2	$0.915\ kgm^2$	$0.853\ kgm^2$	9 %
A_1	$7.17\ kgm^2$	$6.42\ kgm^2$	2.6 %
A_c	$1.085\ kgm^2$	$1.05\ kgm^2$	7 %

Table 8.1: Comparison between the identified and analytical parameters of the model and deviation on the identified parameters for noise measurements.

8.4 Control System Design.

For control design the model (equation 8.1, 8.2 and 8.3) is put in the suitable form:

$$\mathbf{M}(\theta_{l2}) \begin{bmatrix} \ddot{\theta}_{m1} \\ \ddot{\theta}_{l2} \\ \ddot{q}_{j1} \end{bmatrix} + \begin{bmatrix} F_{\theta_{m1}}(\mathbf{x}) \\ F_{\theta_{l2}}(\mathbf{x}) \\ F_{q_{j1}}(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} T_1 \\ T_2 \\ 0 \end{bmatrix} \tag{8.7}$$

with

$$q_{j1} = \theta_{l1} - \theta_{m1}$$

$$\mathbf{M}(\theta_{l2}) = \begin{bmatrix} I_{m1} + A_{11}(\theta_{l2}) & A_{12}(\theta_{l2}) & A_{11}(\theta_{l2}) \\ A_{12}(\theta_{l2}) & A_2 & A_{12}(\theta_{l2}) \\ A_{11}(\theta_{l2}) & A_{12}(\theta_{l2}) & A_{11}(\theta_{l2}) \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{M}_{11}\ (2 \times 2) & \mathbf{M}_{12}\ (2 \times 1) \\ \mathbf{M}_{12}^T\ (1 \times 2) & \mathbf{M}_{22}\ (1 \times 1) \end{bmatrix}$$

where

$$A_{11}(\theta_{l2}) = A_1 + 2\ A_c\ cos(\theta_{l2})$$

$$A_{12}(\theta_{l2}) = A_2 + A_c\ cos(\theta_{l2})$$

and

$$F_{\theta_{m1}}(\mathbf{x}) = C_{m1}\dot{\theta}_{m1} - 2\,A_c\,\dot{\theta}_{l1}\dot{\theta}_{l2}sin(\theta_{l2}) - A_c\,\dot{\theta}_{l2}^2 sin(\theta_{l2})$$
$$F_{\theta_{l2}}(\mathbf{x}) = C_{m2}\dot{\theta}_{l2} + A_c\,\dot{\theta}_{l1}^2 sin(\theta_{l2})$$
$$F_{q_{j1}}(\mathbf{x}) = -2\,A_c\,\dot{\theta}_{l1}\dot{\theta}_{l2}sin(\theta_{l2}) - A_c\,\dot{\theta}_{l2}^2 sin(\theta_{l2}) + K_{j1}q_{j1}$$

The state vector is denoted by:

$$\mathbf{x} = [\theta_{m1}\ \ \theta_{l2}\ q_{j1}\ \dot{\theta}_{m1}\ \dot{\theta}_{l2}\ \dot{q}_{j1}]^T$$

The desired state vector is denoted by:

$$\mathbf{x}_d = [\theta_{m1d}\ \ \theta_{l2d}\ q_{j1d}\ \dot{\theta}_{m1d}\ \dot{\theta}_{l2d}\ \dot{q}_{j1d}]^T$$

The inverse mass matrix $\mathbf{W}$ is given by:

$$\mathbf{W} = \mathbf{M}^{-1} = \begin{bmatrix} \frac{1}{I_{m1}} & 0 & -\frac{1}{I_{m1}} \\ 0 & \frac{A_{11}(\theta_{l2})}{m_{cc}} & -\frac{A_{12}(\theta_{l2})}{m_{cc}} \\ -\frac{1}{I_{m1}} & -\frac{A_{12}(\theta_{l2})}{m_{cc}} & \frac{1}{I_{m1}} + \frac{A_2}{m_{cc}} \end{bmatrix}$$
$$= \begin{bmatrix} \mathbf{W}_{11}\ (2\times 2) & \mathbf{W}_{12}\ (2\times 1) \\ \mathbf{W}_{12}^T\ (1\times 2) & \mathbf{W}_{22}\ (1\times 1) \end{bmatrix}$$

with $m_{cc} = A_2 A_{11}(\theta_{l2}) - A_{12}^2(\theta_{l2})$

8.4.1 The state estimator

The nonlinear controller described in chapter 4 assumes that all the states are available for feedback. However the velocities $\dot{\theta}_{m1}$, $\dot{\theta}_{l2}$ and $\dot{q}_{j1}$ can not be measured directly.

In chapter 5 two solutions to estimate the unmeasurable states are proposed. The first solution is to use a discrete nonlinear state estimator. This estimator is applicable to all classes of systems. However, the only states which cannot be measured directly, are the velocities. Therefore the second proposed solution is to estimate the velocities by differentiation.

In spite of the reduction of computation time, obtained by using this second solution, the discrete nonlinear state estimator is used here.

The purpose is to experimentally evaluate the proposed estimator design.

The design of a nonlinear state estimator is described in chapter 5. It is a current estimator consisting of two stages: a measurement update:

$$\hat{\mathbf{x}}_k = \tilde{\mathbf{x}}_k + \mathbf{L}(\mathbf{y}_k - \tilde{\mathbf{y}}_k)$$

and a time update:

$$\tilde{\mathbf{x}}_{k+1} = \int_{t_k}^{t_{k+1}} f(\hat{\mathbf{x}}_k, T_{1k}, T_{2k})dt$$

with

$$f(\hat{\mathbf{x}}_k, T_{1k}, T_{2k}) = \begin{bmatrix} \hat{x}_{4,k} \\ \hat{x}_{5,k} \\ \hat{x}_{6,k} \\ -\mathbf{M}^{-1}(\hat{\mathbf{x}}_k) \begin{bmatrix} F_{\theta_{m1}}(\hat{\mathbf{x}}_k) \\ F_{\theta_{l2}}(\hat{\mathbf{x}}_k) \\ F_{q_{j1}}(\hat{\mathbf{x}}_k) \end{bmatrix} + \mathbf{M}^{-1}(\hat{\mathbf{x}}_k) \begin{bmatrix} T_{1k} \\ T_{2k} \\ 0 \end{bmatrix} \end{bmatrix}$$

The estimator gain matrix, $\mathbf{L}$, is calculated based on the linearized and discretized state space model. This model is obtained by first linearizing the continuous time model and then transforming it to discrete time according to the zero–order hold equivalence.

The continuous linearized model in state space form is given by equation 4.20, where

$$\frac{\delta \mathbf{F}\mathbf{q}_0}{\delta \mathbf{x}}\Big|_{\mathbf{x}_d} = \begin{bmatrix} \frac{\delta F_{\theta_{m1}}}{\delta \mathbf{x}}\Big|_{\mathbf{x}_d} \\ \frac{\delta F_{\theta_{l2}}}{\delta \mathbf{x}}\Big|_{\mathbf{x}_d} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & C_{m1} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{m2} & 0 \end{bmatrix}$$

$$\frac{\delta \mathbf{F}\mathbf{q}_f}{\delta \mathbf{x}}\Big|_{\mathbf{x}_d} = \frac{\delta F_{q_{j1}}}{\delta \mathbf{x}}\Big|_{\mathbf{x}_d} = [\, 0 \quad 0 \quad K_{j1} \quad 0 \quad 0 \quad 0 \,]$$

if the model is linearized around zero velocity. The estimator gain matrix, $\mathbf{L}$, is calculated using a pole placement algorithm. The continuous time poles are placed at

$$\begin{bmatrix} -2\,\pi\,10 \\ -2\,\pi\,10 \\ -2\,\pi\,10 \pm 2\,\pi\,10\,i \\ -2\,\pi\,15 \pm 2\,\pi\,15\,i \end{bmatrix} \; rad/s$$

They are transformed to discrete time using (zero–order hold sampling)

$$z_i = e^{p_i T_s}$$

with p_i the continuous time poles and T_s the sampling period. Only one constant feedback gain matrix is chosen. The feedback gain matrix is designed with respect to the final desired position.

The integration of the nonlinear model, needed in the time update of the states, is provided by a first order integration rule.

Filtering of the measured outputs and adding the filter dynamics in the model is not needed, when this nonlinear state estimator is used. There is no observation spill–over because of the absence of higher frequency modes which can influence the controller.

Figure 8.4 shows the estimated velocities when the system is excited with a band limited random signal. The cut–off frequency of the signal spectrum is 20 Hz. The estimated velocities are compared with the velocities estimated by a second order differentiator. Similar results are obtained. The figure showing the estimated deformation velocity, $\dot{q}_{j1}$ shows that $\dot{q}_{j1}$ calculated by differentiation is more noisy than $\dot{q}_{j1}$ calculated by the state estimator (no filtering of measurements is used here). The error on the estimated positions is usually smaller than 6 counts on the motor encoders.

8.4.2 The control design

The control torque is given by

$$\mathbf{T} = \mathbf{T}_{ff} + \mathbf{W}_{11}^{-1}\mathbf{T}_{fb} \tag{8.8}$$

where the feedback torque $\mathbf{T}_{fb}$ is given by

$$\begin{bmatrix} T_{fb1} \\ T_{fb2} \end{bmatrix} = -\mathbf{K}(\mathbf{x} - \mathbf{x}_d) \tag{8.9}$$

$\mathbf{K}$ is a 2 x 6 feedback gain matrix.

The control design consists of the calculation of: 1) the feedforward torque $\mathbf{T}_{ff}$, 2) the feedback gain matrix $\mathbf{K}$, and 3) the desired state variables.

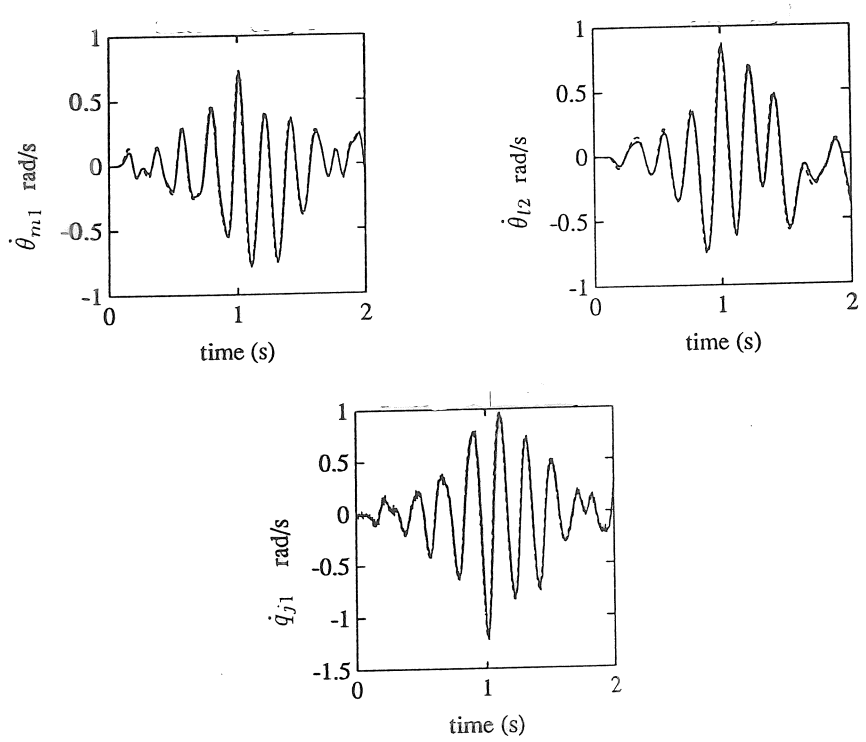

Figure 8.4: Velocity of the first and second motor and deformation velocity calculated by differentiation (solid line) and estimated by the nonlinear state estimator (dashed line). The system is excited with a band limited random signal.

1) The feedforward torque, $\mathbf{T}_{ff}$, is given by equation 4.14

$$\begin{bmatrix} T_{ff1} \\ T_{ff2} \end{bmatrix} = \begin{bmatrix} F_{\theta_{m1}}(\mathbf{x}_d) \\ F_{\theta_{l2}}(\mathbf{x}_d) \end{bmatrix} + \mathbf{W}_{11}^{-1}(\theta_{l2d})\mathbf{W}_{12}(\theta_{l2d})\, F_{q_{j1}}(\mathbf{x}_d) + \mathbf{W}_{11}^{-1}(\theta_{l2d}) \begin{bmatrix} \ddot{\theta}_{m1d} \\ \ddot{\theta}_{l2d} \end{bmatrix} \tag{8.10}$$

or

$$T_{ff1} = F_{\theta_{m1}}(\mathbf{x}_d) - F_{q_{j1}}(\mathbf{x}_d) + I_{m1}\ddot{\theta}_{m1d}$$

$$T_{ff2} = F_{\theta_{l2}}(\mathbf{x}_d) - \frac{A_{12}(\theta_{l2})}{A_{11}(\theta_{l2})}\, F_{q_{j1}}(\mathbf{x}_d) + \left(A_2 - \frac{A_{12}^2(\theta_{l2})}{A_{11}(\theta_{l2})}\right) \ddot{\theta}_{l2d}$$

2) The feedback gain computation is based on the linearized model 4.20, transformed to discrete time according to the zero–order hold equivalence. The feedback gains are calculated using a pole placement algorithm. The best results are obtained with the continuous time poles placed at

$$\begin{bmatrix} -2\,\pi\,1.91 \\ -2\,\pi\,2.33 \\ -2\,\pi\,2.75 \pm 2\,\pi\,2.75\,i \\ -2\,\pi\,3.18 \pm 2\,\pi\,3.18\,i \end{bmatrix} \; rad/s$$

To limit the computation time, the feedback gains are chosen constant: they are computed with respect to the final desired position.

3) The reference values of the state variables

$$\mathbf{x}_d = [\theta_{m1d}\;\; \theta_{l2d}\; q_{j1d}\; \dot{\theta}_{m1d}\; \dot{\theta}_{l2d}\; \dot{q}_{j1d}]^T$$

are calculated based on the nonlinear model, as described in chapter 5. Given are the angular position, velocity and acceleration of each link , θ_{l1d}, θ_{l2d}, $\dot{\theta}_{l1d}$, $\dot{\theta}_{l2d}$, $\ddot{\theta}_{l1d}$, $\ddot{\theta}_{l2d}$, obtained by inverse kinematics of a rigid two link robot (appendix C). The trajectory of the tip is generated based on a ninth order polynomial. As explained in chapter 5, the use of such high order polynomials is needed to avoid discontinuities in the feedforward signal. In addition it limits the excitation of the structure.

The reference values for the deflection at the first joint, q_{j1d} and the joint angle θ_{m1d} are calculated from equation 8.3, using the procedure described in section 5.4.2:

$$\begin{aligned} q_{j1d} = \; & [\, -(A_1 + 2\,A_c\,cos(\theta_{l2d}))\,\ddot{\theta}_{l1} - (A_2 + A_c\,cos(\theta_{l2d}))\,\ddot{\theta}_{l2} \\ & +2\,A_c\,\dot{\theta}_{l1d}\dot{\theta}_{l2d}sin(\theta_{l2d}) + A_c\,\dot{\theta}^2_{l2d}sin(\theta_{l2d})\,]\,/\,K_{j1} \\ \theta_{m1d} = \; & \theta_{l1d} - q_{j1d} \end{aligned}$$

Differentiation gives the derivatives $\dot{\theta}_{m1d}$ and $\dot{q}_{j1d}$.

8.4.3 Reduction of steady state errors and limit cycles caused by nonlinear joint friction

A simple compensation scheme for the nonlinar Coulomb friction is used (chapter 5):

$$\begin{aligned} T_{ti} = \; & T_i + 0.9\,F_{mi}\,sgn(\dot{\theta}_{mi}) && if \;\; |\dot{\theta}_{mi}| > \dot{\theta}^{tr}_{mi} \\ T_{ti} = \; & T_i && if \;\; |\dot{\theta}_{mi}| \leq \dot{\theta}^{tr}_{mi} \end{aligned} \qquad (8.11)$$

with T_{ti} the total control torque i, $\dot{\theta}_{mi}$ the motor position i (θ_{m1} or $\theta_{m2} = \theta_{l2}$), $\dot{\theta}^{tr}_{mi}$ the transitional velocity of motor i, determing the width of the deadband around zero velocity, and T_i defined in equation 8.8.
To reduce the risk of overcompensation, only 90 % of the estimated Coulomb friction is compensated. The main effect of this compensation observed in the experiments is that the steady state error of the tip position is reduced by a factor 8.

Other Coulomb friction compensation schemes are tried out [59]. For example, to avoid the discontinuous jumps in the Coulomb friction canceling torque, defined in equation 8.11, experiments are performed with a Coulomb friction canceling torque defined by

$$\frac{0.9F_{mi}}{\dot{\theta}^{tr}_{mi}}\;\dot{\theta}_{mi} \qquad (8.12)$$

for $|\dot{\theta}_{mi}| \leq \dot{\theta}^{tr}_{mi}$ (figure 8.5). The experiments show that in comparison with the first compensation scheme, 8.11, no further improvements are obtained. Therefore, the first, more simple, solution is selected.

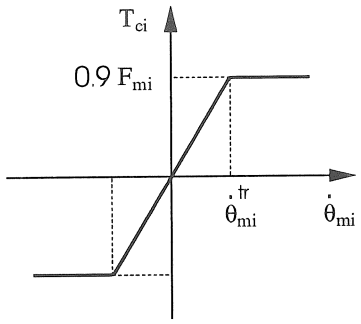

Figure 8.5: Coulomb friction canceling torque

In addition to this Coulomb friction compensation scheme, limit cycles are reduced by phasing out of the feedback once the final position is reached, and steady state errors are eliminated by adding integral action.

8.5 Experimental Results

The experimental results of the controller presented above, are compared to the performance of a controller based on nonlinear decoupling for a rigid robot. The desired motion is a 0.858 m straight line in the x–y plane starting from a point with the manipulator almost completely extended. The motion has to be executed in 1.0 seconds.

8.5.1 Rigid Control

The nonlinear control law is obtained using the dynamic model of a robot without joint flexibility:

$$\mathrm{M}_{11}(\theta_{l2}) \begin{bmatrix} \ddot{\theta}_{l1} \\ \ddot{\theta}_{l2} \end{bmatrix} + \begin{bmatrix} F_{\theta_{l1}}(\theta_{l1}, \theta_{l2}, \dot{\theta_{l1}}, \dot{\theta_{l2}}) \\ F_{\theta_{l2}}(\theta_{l1}, \theta_{l2}, \dot{\theta_{l1}}, \dot{\theta_{l2}}) \end{bmatrix} = \begin{bmatrix} T_1 \\ T_2 \end{bmatrix} \tag{8.13}$$

with $F_{\theta_{l1}}(\theta_{l1}, \theta_{l2}, \dot{\theta_{l1}}, \dot{\theta_{l2}}) = C_{m1}\dot{\theta_{l1}} - 2\,A_c\,\dot{\theta_{l1}}\dot{\theta_{l2}}sin(\theta_{l2}) - A_c\,\dot{\theta}^2_{l2}sin(\theta_{l2})$
The model is obtained from 8.7 by considering only the rigid part of the model, and by setting the motor position θ_{m1} equal to the beam position θ_{l1}. The control torque reduces to:

$$\begin{bmatrix} T_1 \\ T_2 \end{bmatrix} = \begin{bmatrix} F_{\theta_{l1}}(\theta_{l1d}, \theta_{l2d}, \dot{\theta}_{l1d}, \dot{\theta}_{l2d}) \\ F_{\theta_{l2}}(\theta_{l1d}, \theta_{l2d}, \dot{\theta}_{l1d}, \dot{\theta}_{l2d}) \end{bmatrix} + \mathrm{M}_{11}(\theta_{l2d}) \begin{bmatrix} T_{fb1} \\ T_{fb2} \end{bmatrix}$$

$$T_{fb1} = \ddot{\theta}_{l1d} - K_{\theta_{l1}}(\theta_{l1} - \theta_{l1d}) - K_{\dot{\theta}_{l1}}(\dot{\theta}_{l1} - \dot{\theta}_{l1d})$$

$$T_{fb2} = \ddot{\theta}_{l2d} - K_{\theta_{l2}}(\theta_{l2} - \theta_{l2d}) - K_{\dot{\theta}_{l2}}(\dot{\theta}_{l2} - \dot{\theta}_{l2d})$$

The best results are obtained using a controller bandwidth of $1.5Hz$, while the lowest resonance frequency is $4.6Hz$. Figure 8.6 shows the results. The tip position trajectory in the x–y plane is compared with the desired trajectory. In addition a time history of the tip position error normal and tangential to the desired trajectory is plotted. The maximum tracking error normal to the desired trajectory is 300 *mm* while the error tangential to the desired trajectory is more than 500 *mm*. The reason for these large tracking errors is that the large deflection of the springs is not taken into account. The steady state error normal to the desired trajectory is 35 *mm* and the steady state error tangential to the desired trajectory is 18 *mm*. The settling time is 3 seconds.

8.5.2 Generalized nonlinear decoupling

Figure 8.7 shows the results of the generalized nonlinear decoupling controller. The maximum tracking error normal to the desired trajectory is less than 30 *mm* while the error tangential to the desired trajectory is 16 *mm*. The steady state error is highly reduced and is now 1 *mm* in the two directions. After 1.3 seconds the error is less than 2 *mm*. Comparison with figure 8.6 reveals the superior performance of the method.

8.6 Conclusion.

The generalized nonlinear decoupling controller is experimentally verified on a two link robot with a flexible first joint.
Good control performance is obtained after:

- an accurate identification of the nonlinear model,
- inclusion of the nonlinear Coulomb friction,
- trajectory generation which takes the flexibility into account.
- using a nonlinear discrete state estimator.

Comparison with a controller designed for rigid robots shows the superior behaviour of the controller.
In chapter 9 the controller performance of an industrial robot will be improved by taking into account the flexibility in the joints.

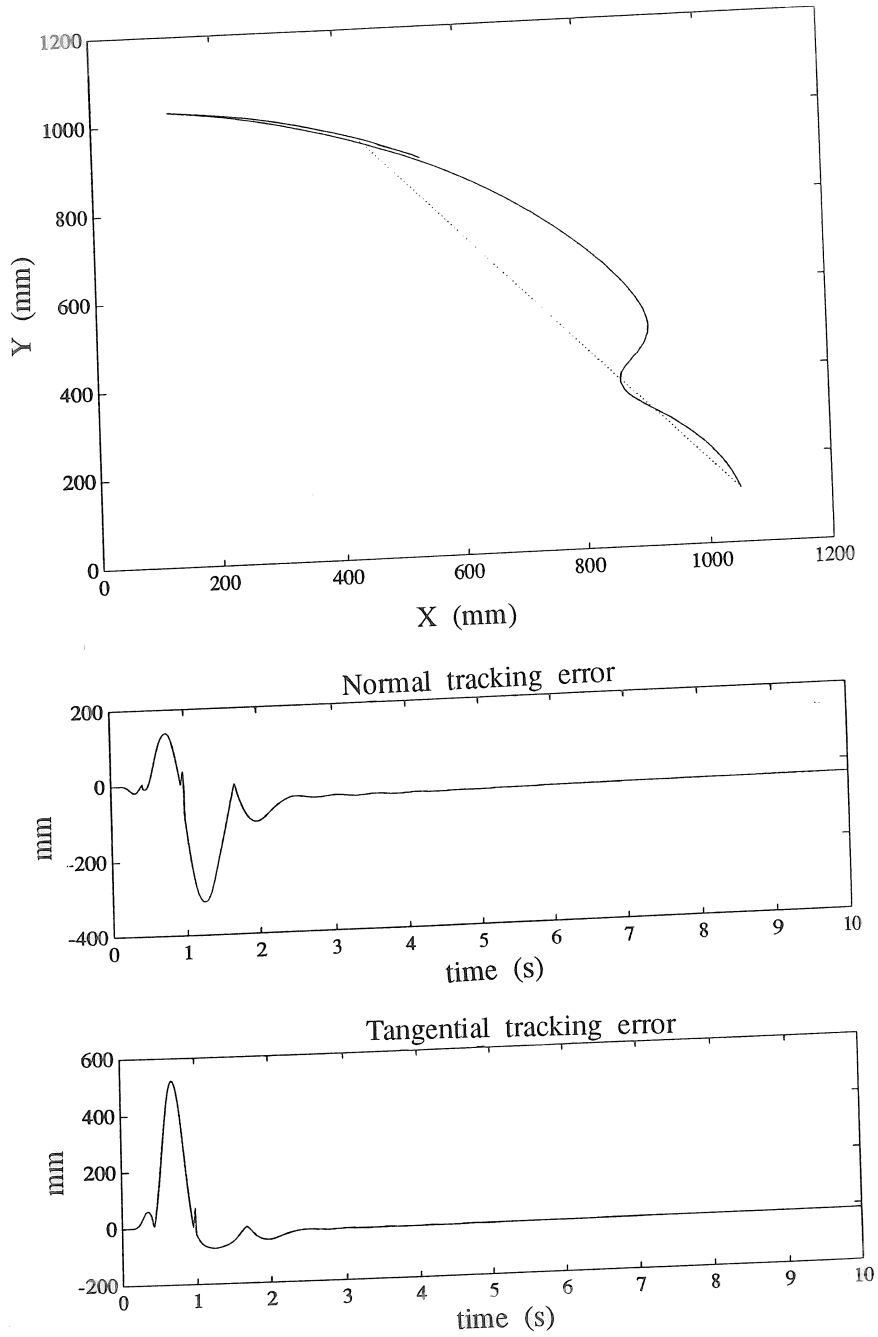

Figure 8.6: Rigid control: The tip position trajectory (solid line) compared with the desired trajectory (dashed line) and the tracking error normal and tangential to the desired trajectory in function of time.

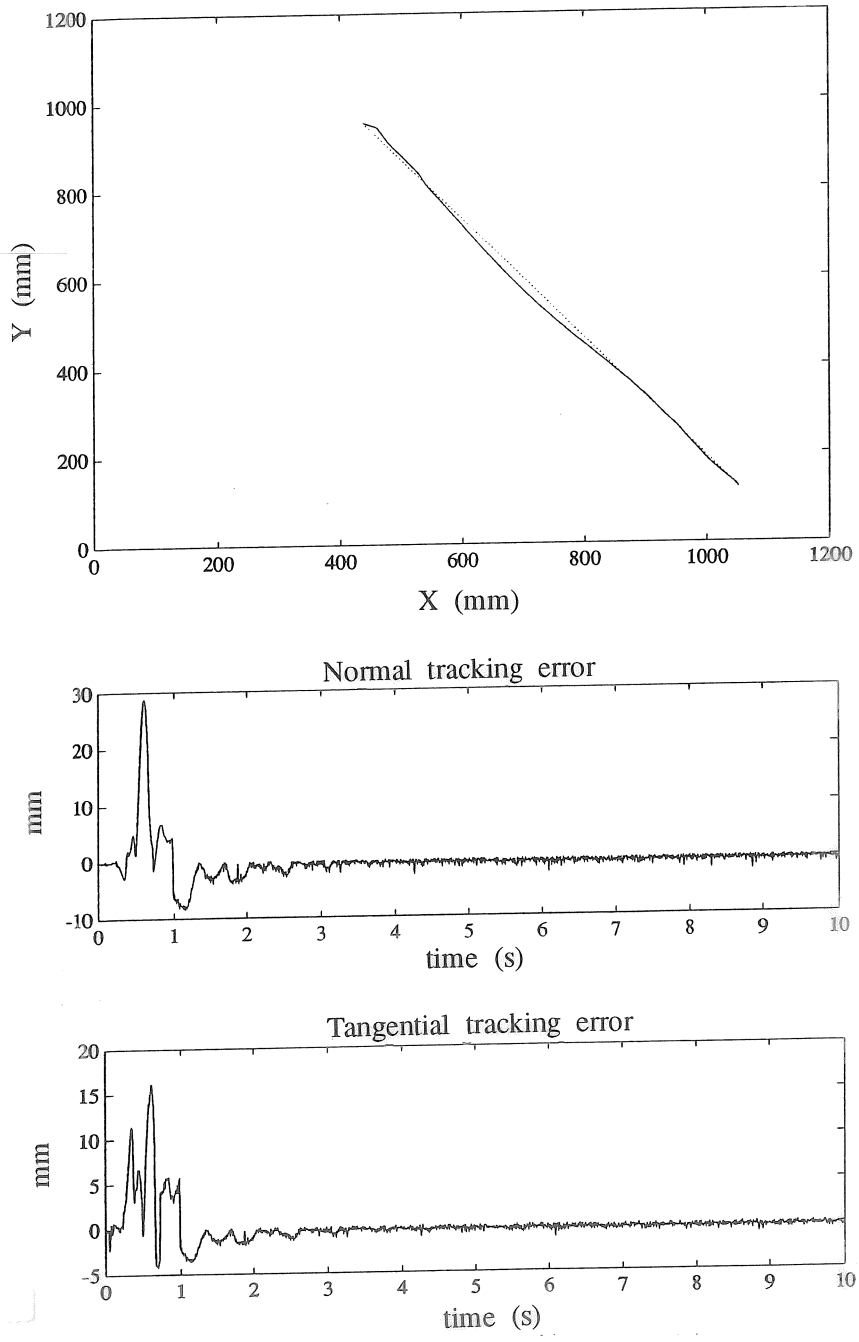

Figure 8.7: Generalized nonlinear decoupling: The tip position trajectory (solid line) compared with the desired trajectory (dashed line) and the tracking error normal and tangential to the desired trajectory in function of time.

Chapter 9

Experimental results on a KUKA IR 161/60 industrial robot

9.1 Introduction

In literature, most control algorithms are tested on idealised and specially developed test setups, because they require an accurate model of the robot. Industrial robots are avoided because they are too far from ideal and therefore difficult to model. In addition flexibility is avoided in industrial robots. This questions the applicability of these algorithms to industrial robots.

In this chapter, the nonlinear flexible control method is implemented on a KUKA IR 161/60 industrial robot. The flexibility of the KUKA robot is limited. The main flexibilities are located in the first three joints. The dynamics of second and third axes are decoupled from the dynamics of the first axis. Therefore, the first axis is controlled using a linear position dependent state feedback controller [68], [4] and the second and third axis are controlled using a nonlinear decoupling controller. The last three joints of the robot are rigid. They are controlled using a PID controller in velocity command.

The experimental results of the flexible controller are compared to the performance of a classical PID controller. There is a significant improvement of a flexible controller over a classical PID controller in terms of tracking accuracy. In addition the static error due to gravity

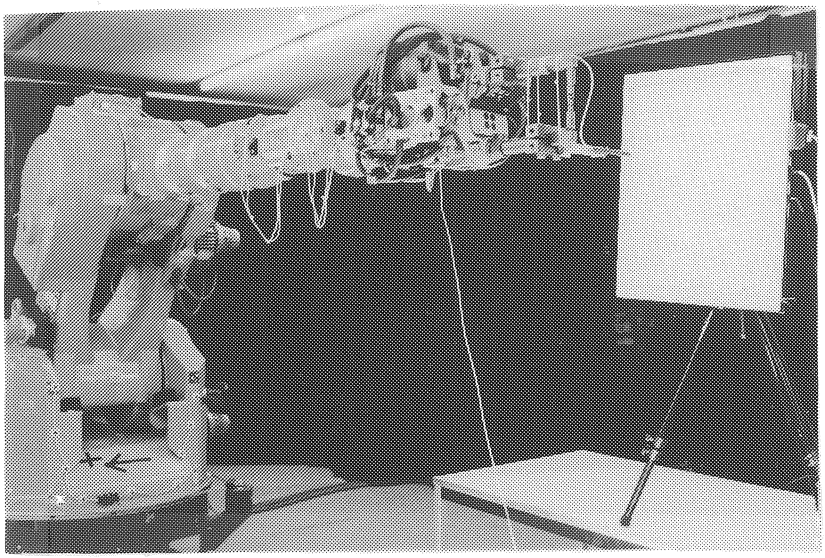

Figure 9.1: Picture of the KUKA IR 161/60 robot in the position of tests, together with the digitizer board.

is compensated. Therefore it is concluded that, even for an industrial robot with limited flexibility, some improvements can be obtained.

Section 9.2 describes the KUKA robot, the sensors, the control hardware and the standard industrial robot controller. Section 9.3 gives the nonlinear model describing the dynamic behaviour of the second and third axis. Section 9.4 describes the identification of the unknown parameters in this model. Section 9.5 describes the control design and section 9.6 reports and interprets the test results of the controller.

9.2 Description of the test setup.

Figure 9.1 shows the setup for the tests. The robot is a KUKA IR 161/60 industrial robot (with an arm extension AV400). It has six axes. It has a working envelope with radius 3150 mm, and it is equiped with a spot welding tool of 45 kg. Table 9.1 gives the characteristics of the second and third motor. Table 9.2 gives some parameters (masses, inertias and distances) of the robot. A compensation for gravity is provided by a hydraulic piston acting on the second link. The hydraulic compensation torque is specified as a linear function of the second link position, θ_{l2}, for a second link position between 0.698 rad (40°) and

2.269 rad (130°):

$$T_{comp} = -3690\,(\frac{\pi}{2} - \theta_{l2}) \tag{9.1}$$

where the second link position is measured with respect to the horizontal axis and the third link position is measured with respect to link 2.

Three extra encoders are attached to the first three links of the robot. They measure the link position in addition to the motor position, which is measured with an encoder mounted on the motor shaft. The difference between the motor encoder signal and the corresponding extra encoder signal is a measure for the deformation of the joint. The initial values of the extra encoders are calculated as the sum of the motor position and the estimated initial static deflection.

An experimental modal analysis of this robot has shown that the main flexibility in this robot is caused by the first three joints [51], such that these extra encoders suffice to measure the flexibility. Figure 9.2 shows the modal analysis results of the first two modes in the position $\theta_{l2} = 0.40\ rad\ (23^o)$, $\theta_{l3} = -0.40\ rad\ (-23^o)$. The results are obtained by measuring frequency response functions of acceleration over force in 135 measurements locations. The force is generated externally by a vibration exciter located on the horizontal part near the tool location. The first mode, with resonance frequency of 5.17 Hz, is a rotation around the vertical axis. The main flexibility is in joint 1. The second mode, with resonance frequency of 7.05 Hz is a flexion in the vertical plane. The flexibility can not be attributed to one single joint: it is a combination of the flexibility in joint 2 and 3.

A digitizing pen and tablet together with the software package RODYM [76] measure the end effector motions and positioning. The digitizing pen is attached to the tool of the robot.

A standard robot controller is delivered with the robot. It controls all axes in velocity command with an independent PID controller. Velocity feedforward is included, and a trajectory based on a trapezoidal velocity (constant acceleration) profile is used.

The dynamics of axes 2 and 3 are decoupled from the dynamics of the first axis. Therefore, the proposed flexible controller controls the first axis with a linear position dependent state feedback controller and the second and third axis with a nonlinear decoupling controller. The last three joints are rigid. They are controlled with a PID controller in velocity command [68], [4]. The tests on the second and third axis are

described in this chapter. For these tests the first and the last three axes are clamped by brakes.

The improved trajectory generation and the control algorithms are programmed on a VME–system. This system can read encoder signals and send out analog voltages in the range of $\pm 10\, volt$. These voltages are linearly converted by the power supply of the motors into a motor current, when the power supply is working in torque command, or a motor velocity, when the power supply is working in velocity command. The command mode can be altered for each axis separately.

9.3 Modeling

The nonlinear equations of motion for axes 2 and 3 are derived applying Lagrange's equations (chapter 2). The second link position is measured with respect to the horizontal axis. The third link position is measured with respect to link 2. The motor friction is modeled as a sum of linear viscous friction and Coulomb friction.
The nonlinear equations of motion are given by:

$$(A_2 + 2A_c\, cos(\theta_{l3}))\, \ddot{\theta}_{l2} + (I_{z3} + A_c\, cos(\theta_{l3}))\, \ddot{\theta}_{l3} + B_2(\theta_{l2}, \theta_{l3}, \dot{\theta}_{l2}, \dot{\theta}_{l3}) + K_{j2}(\theta_{l2} - \frac{\theta_{m2}}{r_2}) = 0 \tag{9.2}$$

$$(I_{z3} + A_c\, cos(\theta_{l3}))\, \ddot{\theta}_{l2} + I_{z3}\, \ddot{\theta}_{l3} + B_3(\theta_{l2}, \theta_{l3}, \dot{\theta}_{l2}, \dot{\theta}_{l3}) + K_{j3}(\theta_{l3} - \frac{\theta_{m3}}{r_3}) = 0 \tag{9.3}$$

$$I_{m2}\ddot{\theta}_{m2} + T_{f2} - \frac{K_{j2}}{r_2}(\theta_{l2} - \frac{\theta_{m2}}{r_2}) = T_2 \tag{9.4}$$

$$I_{m3}\ddot{\theta}_{m3} + T_{f3} - \frac{K_{j3}}{r_3}(\theta_{l3} - \frac{\theta_{m3}}{r_3}) = T_3 \tag{9.5}$$

where $B_2(\theta_{l2}, \theta_{l3}, \dot{\theta}_{l2}, \dot{\theta}_{l3}) = m_3 g x_{3c}\, cos(\theta_{l2} + \theta_{l3}) + (m_3 g l_2 + m_2 g x_{2c})\, cos(\theta_{l2}) - A_c\, \dot{\theta}_{l3}(2\dot{\theta}_{l2} + \dot{\theta}_{l3})\, sin(\theta_{l3}) - T_{comp}$

$B_3(\theta_{l2}, \theta_{l3}, \dot{\theta}_{l2}, \dot{\theta}_{l3}) = m_3 g x_{3c}\, cos(\theta_{l2} + \theta_{l3}) + A_c\, \dot{\theta}_{l2}^{\,2} sin(\theta_{l3})$

and θ_{m2}, θ_{m3} : angle of rotation of second and third actuator rotor

		Axis 2	Axis 3
available torque	Nm	10.3	10.3
maximum torque	Nm	110	110
maximum torque (system limitation)	Nm	33.2	33.2
transmission ratio's		166.8	129.268
torque constant	Nm/A	0.664	0.664
power supply gain	A/V	5	5
damping constant	$10^{-3}\ Nmsec$	4.8	4.8
inertia:			
motor	$10^{-3}\ kgm^2$	9.2	9.2
brakes	$10^{-3}\ kgm^2$	0.95	0.95
tacho	$10^{-3}\ kgm^2$	0.11	0.11
total	$10^{-3}\ kgm^2$	10.5	10.5
encoder resolution	imp/rad	108736.7	84269.83

Table 9.1: Motor characteristics

masses	
link2 :	$m_2 = 225\ kg$
link3 :	$m_{l3} = 230\ kg$
tool :	$m_t = 45\ kg$
link 3 with tool :	$m_3 = 275\ kg$
center of mass position	
link2 :	$x_{2c} = 0.637\ m$
link3 :	$x_{l3c} = 0.049\ m$
link 3 with tool :	$x_{3c} = 0.311\ m$
distances between joints	
link2 :	$l_2 = 0.9728\ m$
link3 :	$l_{l3} = 1.65\ m$
link 3 until endpoint :	$l_{l3} = 2.1233\ m$
moments of inertia with respect to axis	
link2 :	$I_{z2} = 120\ kgm^2$
link3 :	$I_{zl3} = 90\ kgm^2$
link 3 with tool :	$I_{z3} = 212.5\ kgm^2$
motor and drive (reduced to motor)	
motor 2 :	$I_{m2} = 0.013\ kgm^2$
motor 3 :	$I_{m3} = 0.013\ kgm^2$
Calculated model parameters	
$A_2 = I_{z2} + I_{z3} + m_3 l_2^2 = 593 kgm^2$	
$A_c = m_3 l_2 x_{3c} = 83.2 Nm$	
$m_3 g x_{3c} = 839 Nm$	
$m_3 g l_2 + m_2 g x_{2c} = 4030 Nm$	

Table 9.2: Parameters specified by KUKA or calculated from the specifications

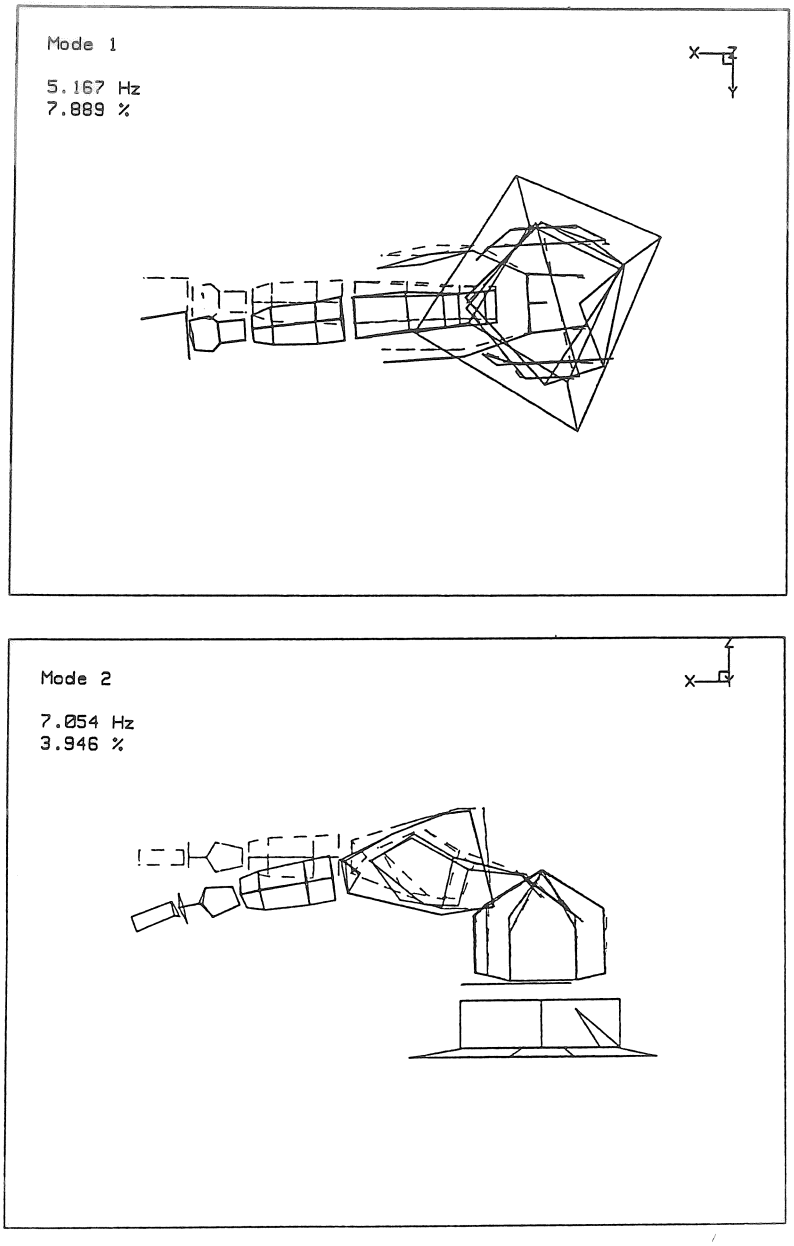

Figure 9.2: Modal analysis results: first two modes in the position $\theta_{l2} = 0.40\ rad\ (23^o)$, $\theta_{l3} = -0.40\ rad\ (-23^o)$

θ_{l2}, θ_{l3}	: angular position of second and third link
T_2, T_3	: motor torque 2 and 3
r_2, r_3	: transmission gear ratio 2 and 3
I_{m2}, I_{m3}	: rotor inertia 2 and 3
I_{z3}	: moment of inertia of third link
T_{f2}, T_{f3}	: motor friction torque 2 and 3
K_{j2}, K_{j3}	: spring constant 2 and 3
A_2	$= I_{z2} + I_{z3} + m_3 l_2^2$
A_c	$= m_3 l_2 x_{3c}$
T_{comp}	$= -3690\ (\frac{\pi}{2} - \theta_{l2})$: torque compensating the gravity
m_i	: ith link mass
x_{ic}	: ith center of mass position

9.4 Identification

Accurate knowledge of the parameters in the nonlinear model is important, since the achievable performance of the controller (especially the tracking performance) is limited by the modeling accuracy. In many publications control is either simulated, or applied to a specially designed test set up with exactly known characteristics. For an industrial robot, however, identification of the parameters is needed. Some parameters can be found in engineering drawings or can be calculated from geometrical properties. For the KUKA robot, these values are summarized in table 9.2. The obtained results are not always reliable because of the complex geometries of industrial robots. Identification of these parameters is useful to validate or improve the calculated values. Other parameters, like the spring constants, have to be identified experimentally.

Practice shows that it is not a good approach to try to identify all the parameters in one step. It is necessary to split up the identification problem in several smaller subproblems, and to use different identification techniques. Therefore, in this section, the identification is performed in two steps: First, the parameters related to the rigid body model are estimated. Then the spring constants are estimated using static, quasi–static, noise and stepped sine measurements. Comparison of the results gives an idea of the obtained accuracy.

9.4.1 Identification of inertial, gravitational and friction coefficients.

The same identification procedure as in chapter 8 is used. The method starts directly from the nonlinear equations of motion. Under infinite stiffness assumptions the model 9.2–9.5 reduces to:

$$(I_{m2}+A_2/r_2^2+2A_c/r_2^2\ cos(\theta_{l3}))\ \ddot{\theta}_{m2}+(I_{z3}/r_2r_3+A_c/r_2r_3\ cos(\theta_{l3}))\ \ddot{\theta}_{m3}$$
$$+ T_{f2} + B_2(\theta_{l2}, \theta_{l3}, \dot{\theta}_{l2}, \dot{\theta}_{l3})/r_2 = T_2 + T_{comp} \qquad (9.6)$$
$$(I_{z3}/r_2r_3 + A_c/r_2r_3\ cos(\theta_{l3}))\ \ddot{\theta}_{m2} + (I_{m3} + I_{z3}/r_3^2)\ \ddot{\theta}_{m3}$$
$$+ T_{f3} + B_3(\theta_{l2}, \theta_{l3}, \dot{\theta}_{l2}, \dot{\theta}_{l3})/r_3 = T_3 \qquad (9.7)$$

Known input torques, T_2 and T_3, are applied to the system. The corresponding angle of rotation of the actuator rotors and corresponding angular position of the links are measured. Introducing the measured values and their first and second derivatives in the non linear equations gives rise to an overdetermined set of linear equations. Solving this set of equations in a linear least squares sense gives estimates for the unknown parameters.

The identification method requires motion of a single axis, while the positions of the others are fixed. A test motion with high accelerations but small changes in position is chosen. In this way the coupling torques acting on the moving axis are small, and the gravitational forces can be regarded as constant. The main component of the motor torque can thus be related to the effective inertia of the moving axis and to the constant gravitational forces. The coupling inertias are determined from the same measurements by considering the control torques needed to maintain the fixed position at the other axes; If a joint j of the robot is accelerating, a coupling force will appear on axis i. The main component of the motor torque i, needed to maintain the fixed position can thus be related to the coupling inertia and to the gravitational forces.

By repeating the procedure in different positions, sets of unknown inertial and gravitational coefficients are obtained as a function of the joint position.

The position dependent inertial coefficients are given by:

$$I_{m2} + \frac{A_2}{r_2^2} + 2\frac{A_c}{r_2^2}cos(\theta_{l2})$$

$$I_{m3} + \frac{I_{z3}}{r_3^2}$$

$$\frac{I_{z3}}{r_2 r_3} + \frac{A_c}{r_2 r_3} cos(\theta_{l2})$$

From this set of coefficients the parameters A_2, A_c, I_{z3}, I_{m2} and I_{m3} are solved. Equality of the two motor inertias is assumed.

The position dependent gravitational forces are given by

$$m_3 g x_{3c}\, cos(\theta_{l2} + \theta_{l3}) + (m_3 g l_2 + m_2 g x_{2c})\, cos(\theta_{l2})$$

for the second link, and by

$$m_3 g x_{3c}\, cos(\theta_{l2} + \theta_{l3})$$

for the third link. The two factors $m_3 g x_{3c}$ and $m_3 g l_2 + m_2 g x_{2c}$ are considered as two parameters in the model and are identified from these gravitational forces.

Accurate identification of the motor friction is difficult to perform. For specially designed test set ups, measurements on the direct drive motors can be performed on separate set ups as described in chapter 8. This is not possible for an industrial robot.
Therefore the Coulomb friction coefficient of motor i, F_{mi}, is estimated using the following procedure: The motor torque T_i is gradually changed. The motor torque T_i^+ and T_i^- that is needed to start a motion in the positive and negative direction is measured. Then F_{mi} is estimated as

$$F_{mi} = \frac{T_i^+ - T_i^-}{2}$$

In later identification procedures, the Coulomb friction torque $F_{mi} sgn(\dot{\theta}_{mi})$, is compensated in the excitation signal. In other words, the motor torques used in the identification are the applied excitation signals reduced by the Coulomb friction torques. In this way the Coulomb friction is implicitely taken into account in the identification.
The linear motor friction coefficients are determined from the noise measurements described in section 9.4.2.

The identified parameters are summarized in table 9.3.
The applied motions are motions of axis 2 resp. 3 of 5^o in 1 seconds,

		analytical	quasi–static
A_2	(kgm^2)	597	576
A_c	(kgm^2)	91.3	115
I_{z3}	(kgm^2)	214	199
I_{m2}	(kgm^2)	0.013	0.0158
I_{m3}	(kgm^2)	0.013	0.0158
$m_3 g x_{3c}$	(Nm)	835	782
$m_3 g l_2 + m_2 g x_{2c}$	(Nm)	4030	4090

Table 9.3: Comparison between the identified and analytical parameters of the rigid body model

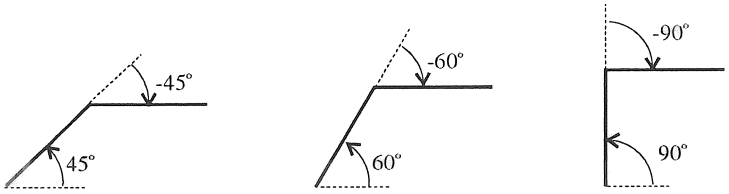

Figure 9.3: Identification of inertial, gravitational and friction coefficients; measurement positions

while the position of the other axis is fixed. The measurements are performed in three different positions (figure 9.3);
$\theta_{l2} = \frac{\pi}{4}\ rad\,(45^o)$ and $\theta_{l3} = -\frac{\pi}{4}\ rad\,(-45^o)$,
$\theta_{l2} = \frac{\pi}{3}\ rad\,(60^o)$ and $\theta_{l3} = -\frac{\pi}{3}\ rad\,(-60^o)$ and,
$\theta_{l2} = \frac{\pi}{2}\ rad\,(90^o)$ and $\theta_{l3} = -\frac{\pi}{2}\ rad\,(-90^o)$.
Figure 9.4 gives the measured and estimated torques in the position $\theta_{l2} = \frac{\pi}{4}\ rad\,(45^o)$, $\theta_{l3} = -\frac{\pi}{4}\ rad\,(-45^o)$.
The motion is obtained by applying a PD controller with appropriate feedforward. The feedforward calculation is based on the analytical parameters.

The identification process is repeated for different measurements in the same position. Comparing the identification results shows that there is still a deviation of 6% on the identified gravitational constants. This is caused by the high influence of the nonlinear Coulomb friction on the identification.

The gravitational constant $m_3 g x_{3c}$ can also be determined from a slow motion of the third axis over a high position range. The position

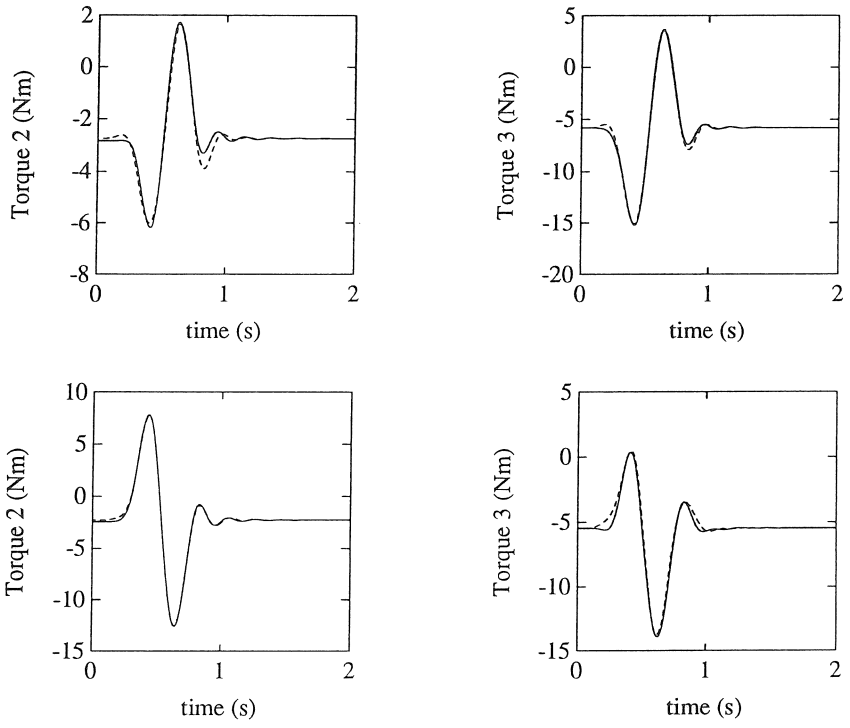

Figure 9.4: Measured (solid line) and estimated (dashed line) torques of the second and third axis for a position step of 5^o for link 2 (upper figures), resp. link 3 (bottom figures).

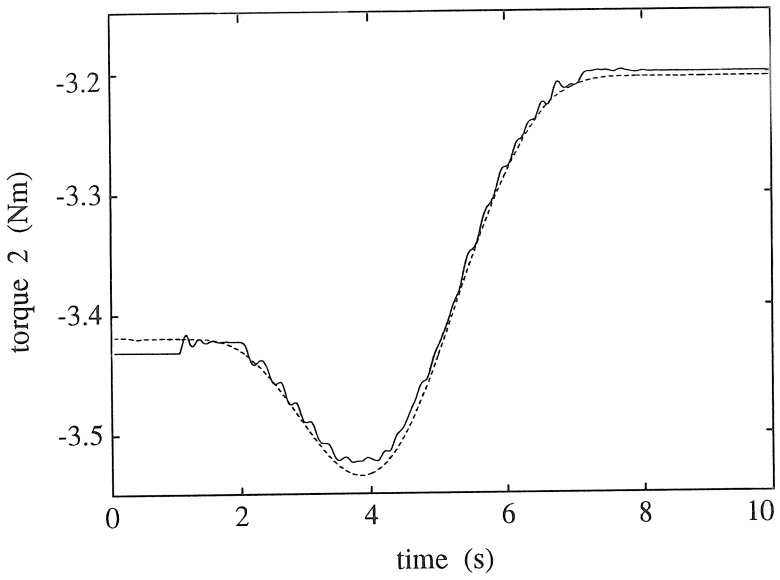

Figure 9.5: Measured (solid line) and estimated (dashed line) torque 2 for a slow movement of the third axis.

of the second link is kept constant. The controller of the second axis compensates for the changes in gravitation:

$$T_2 = \frac{m_3 g x_{3c}}{r2} cos(\theta_{l2} + \theta_{l3})$$

Figure 9.5 shows the measured and identified torque for this measurement on the KUKA IR 161/60 robot.

The obtained Coulomb friction coefficient is 0.670 Nm for motor 2 and 0.487 Nm for motor 3. The obtained linear friction coefficient is 0.006 Nms for motor 2 and 0.005 Nms for motor 3. The friction coefficients are optimized by trial and error in the control experiments. There is a good correspondence with the identified values.

9.4.2 Identification of the spring constants.

The spring constants are estimated using static, quasi–static, noise and stepped sine measurements.

- Static measurements: In four different positions (the third link always horizontal), the endpoint (without spot welding tool) is

loaded with a mass $m_l = 96.7\ kg$. The applied torque is calculated as:

$$T_{p2} = gm_l l_3\ cos(\theta_{l2} + \theta_{l3}) + gm_l l_2\ cos(\theta_{l3})$$

$$T_{p3} = gm_l l_3\ cos(\theta_{l2} + \theta_{l3})$$

The deformation before and after loading is measured with three different measurement devices:

- The difference between motor encoder and extra encoder,
- An autocollimator which measures the deflection before and after the third joint. The first measurement gives the deflection of the second joint. The difference between the two measurements gives the deflection of the third joint.
- A comparator which measures the total deflection of the endpoint.

The deflection of the third joint measured by the autocollimator is 22 % higher than the deflection measured using the extra encoder. For the second joint the two measurement devices give the same results. The total deflection at the endpoint measured by the comparator is 25% higher than measured by the extra encoders. A conclusion is that the extra encoders do not measure the whole deflection of the endpoint. This is explained by the fact that there is also some link deflection.

From the difference between the deformations before and after loading, Δq_{j2} and Δq_{j3}, the spring constants are calculated:

$$K_{j2} = -\frac{T_{p2}}{\Delta q_{j2}}$$

$$K_{j3} = -\frac{T_{p3}}{\Delta q_{j3}}$$

- Quasi-static measurements: The same measurement data as in 9.4.1 are used. The deflection $q_{j2} = \theta_{l2} - \frac{\theta_{m2}}{r_2}$ and $q_{j3} = \theta_{l3} - \frac{\theta_{m3}}{r_3}$ is considered. K_{j2} and K_{j3} are calculated from the relationship (equation 9.4 and 9.5)

$$\Delta q_{j2} = -\frac{r_2}{K_{j2}}(\Delta T_2 - I_{m2}\ddot{\theta}_{m2})$$

$$\Delta q_{j3} = -\frac{r_3}{K_{j3}}(\Delta T_3 - I_{m3}\ddot{\theta}_{m3})$$

where Δq_{j2} and Δq_{j3} are the variations in deflection with respect to the initial point, and ΔT_2 and ΔT_3 are the variations in motor torque with respect to the initial point. Considering only variations with respect to the initial point is needed because the inital values of the extra encoders are not known. By considering only variations with respect to the initial point, the influence of the extra encoder offsets is eliminated. Once an accurate estimate of the spring constants is obtained, the initial values of the extra encoders can be calculated as the sum of the motor position and the estimated initial static deflection.
Figure 9.6 gives the measured and identified deflection for each joint in the position $\theta_{l2} = \frac{\pi}{4}\ rad\ (45^o)$, $\theta_{l3} = -\frac{\pi}{4}\ rad\ (-45^o)$.
The same results are obtained by identification of the spring constants from equation 9.2 and 9.3 .

- Band limited noise measurements: The spring constants are determined from measured data obtained by applying bandlimited noise to the second and third motor separately. A constant torque compensating the gravity is added. A more detailed description of this identification method was given in chapter 8 : "identification of a two link test structure with a flexible first joint". Considering the yet identified parameters (inertial, gravitational and friction coefficients) as known parameters improves the identification results.

- Transfer function measurements: Transfer functions are measured using stepped sine measurements in the position $\theta_{l2} = 0.489\ rad\ (28°)$, $\theta_{l3} = -0.489\ rad\ (-28°)$. The excitation is generated by the second and third motor separately while the other motors are clamped by brakes. The measured resonance and anti–resonance frequencies are summarized in table 9.4. Analytical expressions for the resonance and anti–resonance frequencies are derived. The anti–resonance frequency of the transfer function relating the second motor torque to the second motor position, ω_z, is equal to the anti–resonance frequency of the transfer function relating the third motor torque to the third

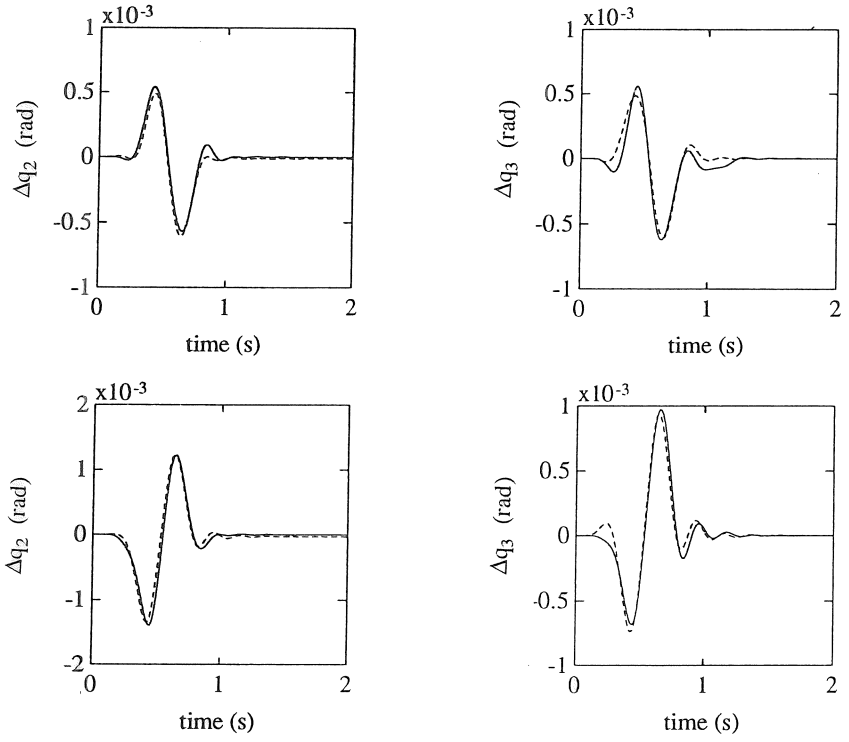

Figure 9.6: Measured (solid line) and identified (dashed line) deflection for joint 2 and 3 for a position step of 5^o for link 2 (upper figures), resp. link 3 (bottom figures).

excitation motor 2:	
second motor torque to second motor position:	
first anti–resonance frequency (ω_z):	6.7 Hz
second motor torque to second joint deflection:	
first resonance frequency:	6.5 Hz
third link position to second link position:	
first anti–resonance frequency (ω_t):	8.4 Hz
excitation motor 3:	
third motor torque to third motor position:	
first anti–resonance frequency (ω_z):	6.6 Hz
first resonance frequency (ω_p):	8.0 Hz
third motor torque to third joint deflection:	
first resonance frequency (ω_p):	7.9 Hz

Table 9.4: Measured resonance and anti–resonance frequencies of the KUKA IR 161/60 industrial robot in the position $\theta_{l2} = 28°$, $\theta_{l3} = -28°$

motor position and is given by:

$$\omega_z^2 = \frac{m_{cf} - \sqrt{m_{cf}^2 - 4m_{cc}K_{j2}K_{j3}}}{2m_{cc}} \tag{9.8}$$

where

$$m_{cc} = A_{22}I_{z3} - A_{23}^2$$
$$m_{cf} = A_{22}K_{j3} + K_{j2}I_{z3}$$
$$A_{22} = A_2 + 2A_c cos(\theta_{l3})$$
$$A_{23} = I_{z3} + A_c cos(\theta_{l3})$$

(ω_z is also equal to the first resonance frequency measured with motor two and three clamped by brakes (figure 9.2).) The resonance frequency of the transfer function relating the third motor torque to the third motor position, ω_p, is equal to the resonance frequency of the transfer function relating the third motor torque to the third joint deflection and is given by:

$$\omega_p^2 = \frac{m_{cp} - \sqrt{m_{cp}^2 - 4m_{cc}I_{m3}K_{j2}K_{j3}(I_{m3} + \frac{I_{z3}}{r_3^2})}}{2I_{m3}m_{cc}} \tag{9.9}$$

where $m_{cp} = I_{m3}m_{cf} + \frac{K_{j3}m_{cc}}{r_3^2}$.
The first anti–resonance frequency of the transfer function relating the third link position to the second link position, ω_t, is given by:

$$\omega_t = \sqrt{\frac{K_{j3}}{I_{z3}}} \tag{9.10}$$

The derivation of these expressions is based on the linearized model (equations 9.2–9.5), where a motor position is considered as constant if the corresponding motor is clamped. In practice, in these equations, the nonlinear terms, $B_2(\theta_{l2}, \theta_{l3}, \dot{\theta}_{l2}, \dot{\theta}_{l3})$ and $B_3(\theta_{l2}, \theta_{l3}, \dot{\theta}_{l2}, \dot{\theta}_{l3})$, can be neglected:

- The velocities are zero in the linearization point ($\theta_{l20} = 0.489\ rad\ (28°)$, $\theta_{l30} = -0.489\ rad\ (-28°)$, $\dot{\theta}_{l20} = 0$, $\dot{\theta}_{l30} = 0$). Terms in products or squares of the velocities can thus be omitted.
- $\theta_{l2} + \theta_{l3}$ is equal to zero in linearization point. After linearization, the gravitational term $m_3 g x_{3c}\ cos(\theta_{l2} + \theta_{l3})$ can be omitted.
- After linearization, the term

  $$\frac{\delta B_2}{\delta \theta_{l2}} \Delta\theta_{l2} = (-(m_3 g l_2 + m_2 g x_{2c}) sin(\theta_{l20}) - 3690)\Delta\theta_{l2}$$

  is small in comparison with the term $K_{j2}\Delta\theta_{l2}$ and can thus be neglected. (For the KUKA IR 161/60 robot the spring constant K_{j2} is more than 100 times higher than $-(m_3 g l_2 + m_2 g x_{2c}) sin(\theta_{l20}) - 3690$.)

The spring constants are determined by comparing the analytical expressions of the resonance and anti–resonance frequencies with the measured resonance and anti–resonance frequencies. They are function of the two spring constants and are position dependent. An exception is the anti–resonance frequency, ω_t, of the transfer function relating the second link position to the third link position with the third motor clamped. This frequency is position independent and only function of the third spring constant K_{j3}, and is thus well suited for the calculation of this parameter.

In table 9.5, the identified parameters are compared with the analytical parameters. The values obtained by performing static measurements differ from the other identification results. The high values of the spring constants can be explained by the fact that in these measurements the motors are clamped by brakes. It can be concluded that the whole compliance of the joints is not measured if the brakes are on.

For control design, the quasi-static values are used. Using these values, the smallest differences between feedforward and actual torques and between measured and calculated state variables are obtained.

		analytical	quasi-static	noise	stepped sine	static
A_2	(kgm^2)	597	576	–	–	–
A_c	(kgm^2)	91.3	115	–	–	–
I_{z3}	(kgm^2)	214	199	217	–	–
I_{m2}	(kgm^2)	0.013	0.0158	0.0126	–	–
I_{m3}	(kgm^2)	0.013	0.0158	0.00899	–	–
$m_3 g x_{3c}$	(Nm)	835	782	779	–	–
$m_3 g l_2 + m_2 g x_{2c}$	(Nm)	4030	4090	4090	–	–
K_{j2}	$(10^6 \frac{Nm}{rad})$	–	1.35	1.328	1.722	2.90
K_{j3}	$(10^6 \frac{Nm}{rad})$	–	0.612	0.582	0.626	1.56

Table 9.5: Comparison between the identified and analytical parameters of the model.

9.5 Control Design

For nonlinear control design the model is put in the suitable form (chapter 2):

$$\mathbf{M}(\theta_{l3}) \begin{bmatrix} \ddot{\theta}_{m2} \\ \ddot{\theta}_{m3} \\ \ddot{q_{j2}} \\ \ddot{q_{j3}} \end{bmatrix} + \begin{bmatrix} F_{\theta_{m2}}(\mathbf{x}) \\ F_{\theta_{m3}}(\mathbf{x}) \\ F_{q_{j2}}(\mathbf{x}) \\ F_{q_{j3}}(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} T_2 \\ T_3 \\ 0 \\ 0 \end{bmatrix} \qquad (9.11)$$

with

$$\mathbf{x} = [\theta_{m2}\ \ \theta_{m3}\ \ q_{j2}\ \ q_{j3}\ \ \dot{\theta}_{m2}\ \ \dot{\theta}_{m3}\ \ \dot{q}_{j2}\ \ \dot{q}_{j3}]^T$$

$$\mathrm{M}(\theta_{l3}) = \begin{bmatrix} I_{m2} + \frac{A_{22}(\theta_{l3})}{r_2^2} & \frac{A_{23}(\theta_{l3})}{r_2 r_3} & \frac{A_{22}(\theta_{l3})}{r_2} & \frac{A_{23}(\theta_{l3})}{r_2} \\ \frac{A_{23}(\theta_{l3})}{r_2 r_3} & I_{m3} + \frac{I_{z3}}{r_3^2} & \frac{A_{23}(\theta_{l3})}{r_3} & \frac{I_{z3}}{r_3} \\ \frac{A_{22}(\theta_{l3})}{r_2} & \frac{A_{23}(\theta_{l3})}{r_3} & A_{22}(\theta_{l3}) & A_{23}(\theta_{l3}) \\ \frac{A_{23}(\theta_{l3})}{r_2} & \frac{I_{z3}}{r_3} & A_{23}(\theta_{l3}) & I_{z3} \end{bmatrix}$$

$$= \begin{bmatrix} \mathrm{M}_{11}\ (2\times 2) & \mathrm{M}_{12}\ (2\times 2) \\ \mathrm{M}_{12}^T\ (2\times 2) & \mathrm{M}_{22}\ (2\times 2) \end{bmatrix}$$

$$\begin{aligned} F_{\theta_{m2}} &= B_2(\theta_{l2}, \theta_{l3}, \dot{\theta}_{l2}, \dot{\theta}_{l3})/r_2 + T_{f2} \\ F_{\theta_{m3}} &= B_3(\theta_{l2}, \theta_{l3}, \dot{\theta}_{l2}, \dot{\theta}_{l3})/r_3 + T_{f3} \\ F_{q_{j2}} &= B_2(\theta_{l2}, \theta_{l3}, \dot{\theta}_{l2}, \dot{\theta}_{l3}) + K_{j2}\, q_{j2} \\ F_{q_{j3}} &= B_3(\theta_{l2}, \theta_{l3}, \dot{\theta}_{l2}, \dot{\theta}_{l3}) + K_{j3}\, q_{j3} \end{aligned}$$

and

$$\begin{aligned} A_{22}(\theta_{l3}) &= A_2 + 2\, A_c\, cos(\theta_{l3}) \\ A_{23}(\theta_{l3}) &= I_{z3} + A_c\, cos(\theta_{l3}) \end{aligned}$$

The inverse mass matrix W is given by:

$$\mathrm{W} = \mathrm{M}^{-1} = \begin{bmatrix} \frac{1}{I_{m2}} & 0 & -\frac{1}{I_{m2} r_2} & 0 \\ 0 & \frac{1}{I_{m3}} & 0 & -\frac{1}{I_{m3} r_3} \\ -\frac{1}{I_{m2} r_2} & 0 & \frac{I_{z3}}{m_{cc}} + \frac{1}{I_{m2} r_2^2} & -\frac{A_{23}}{m_{cc}} \\ 0 & -\frac{1}{I_{m3} r_3} & -\frac{A_{23}}{m_{cc}} & \frac{A_{22}}{m_{cc}} + \frac{1}{I_{m3} r_3^2} \end{bmatrix}$$

where $m_{cc} = A_{22} I_{z3} - A_{23}^2$

The first stage of the generalized nonlinear decoupling controller is a nonlinear feedback or feedforward to cancel out nonlinear terms

in the rigid body dynamics. The implemented controller uses a nonlinear feedforward. This means that the nonlinear decoupling terms are calculated off–line based on the desired state variables instead of the actual values. The feedforward torque $\mathbf{T}_{ff}$ is given by (equation 4.14):

$$\begin{aligned}\mathbf{T}_{ff} &= \begin{bmatrix} F_{\theta_{m2}} \\ F_{\theta_{m3}} \end{bmatrix} + \mathbf{W}_{11}^{-1}\mathbf{W}_{12} \begin{bmatrix} F_{q_{j2}} \\ F_{q_{j3}} \end{bmatrix} + \mathbf{W}_{11}^{-1} \begin{bmatrix} \ddot{\theta}_{m2d} \\ \ddot{\theta}_{m3d} \end{bmatrix} \\ &= \begin{bmatrix} F_{\theta_{m2}} - \frac{F_{q_{j2}}}{r_2} + I_{m2}\ddot{\theta}_{m2d} \\ F_{\theta_{m3}} - \frac{F_{q_{j3}}}{r_3} + I_{m3}\ddot{\theta}_{m3d} \end{bmatrix}\end{aligned} \tag{9.12}$$

or

$$\begin{aligned}T_{2ff} &= I_{m2}\ddot{\theta}_{m2d} + T_{f2} - \frac{K_{j2}}{r_2} q_{j2d} \\ &= I_{m2}\ddot{\theta}_{m2d} + T_{f2} \\ &\quad -(A_{22}\ddot{\theta}_{l2d} + A_{23}\ddot{\theta}_{l3d} + B_2(\theta_{l2d}, \theta_{l3d}, \dot{\theta}_{l2d}, \dot{\theta}_{l3d}))/r_2\end{aligned}$$

$$\begin{aligned}T_{3ff} &= I_{m3}\ddot{\theta}_{m3d} + T_{f3} - \frac{K_{j3}}{r_3} q_{j3d} \\ &= I_{m3}\ddot{\theta}_{m3d} + T_{f3} \\ &\quad -(A_{23}\ddot{\theta}_{l2d} + I_{z3}\ddot{\theta}_{l3d} + B_3(\theta_{l2d}, \theta_{l3d}, \dot{\theta}_{l2d}, \dot{\theta}_{l3d}))/r_3\end{aligned}$$

The feedforward torques for a nonlinear rigid controller are obtained by putting $\theta_{l2d} = \frac{\theta_{m2d}}{r_2}$ and $\theta_{l3d} = \frac{\theta_{m3d}}{r_3}$ in these equations.

The second stage of the generalized nonlinear decoupling controller is a linear feedback of all state variables. In order to use linear techniques for feedback gains computation, the model is linearized to the first order about the desired trajectory (θ_{l20}, θ_{l30}). The linearized model in state space form is given by equation 4.20 where

$$\frac{\delta F_{\theta_{m2}}}{\delta \mathbf{x}}\Big|_{\mathbf{x}_d} = [\frac{lt_1}{r_2^2} \quad \frac{lt_2}{r_3 r_3} \quad \frac{lt_1}{r_2} \quad \frac{lt_2}{r_2} \quad C_{m2} \quad 0 \quad 0 \quad 0]$$

$$\frac{\delta F_{\theta_{m3}}}{\delta \mathbf{x}}\Big|_{\mathbf{x}_d} = [\frac{lt_2}{r_2 r_3} \quad \frac{lt_2}{r_3^2} \quad \frac{lt_2}{r_3} \quad \frac{lt_2}{r_3} \quad 0 \quad C_{m3} \quad 0 \quad 0]$$

$$\frac{\delta F_{q_{j2}}}{\delta \mathbf{x}}\Big|_{\mathbf{x}_d} = [\frac{lt_1}{r_2} \quad \frac{lt_2}{r_3} \quad lt_1 + K_{j2} \quad lt_2 \quad 0 \quad 0 \quad 0 \quad 0]$$

$$\frac{\delta F_{q_{j3}}}{\delta \mathbf{x}}\Big|_{\mathbf{x}_d} = [\frac{lt_2}{r_2} \quad \frac{lt_2}{r_3} \quad lt_2 \quad lt_2 + K_{j3} \quad 0 \quad 0 \quad 0 \quad 0]$$

and
$lt_1 = -m_3 g x_{3c}\, sin(\theta_{l20} + \theta_{l30}) - (m_3 g l_2 + m_2 g x_{2c})\, sin(\theta_{l20})$
$lt_2 = -m_3 g x_{3c}\, sin(\theta_{l20} + \theta_{l30})$
C_{m2}, C_{m3} : linear viscous friction coefficient of motor 2 and 3

The feedback gains are calculated in discrete time. A sample period of 6 ms is used in the experiments. To limit the computation time, the feedback gains are chosen constant: they are computed with respect to the final desired position. Optimal control theory is used (chapter 3, section 3.4.2.1). The weighting matrices Q and R are tuned so that the desired closed loop dynamics is obtained. More weight is given to the rigid coordinates than to the flexible coordinates. The weighting matrices are diagonal matrices given by:

$$Q = \begin{bmatrix} 15 & & & & & & & \\ & 15 & & & & & & \\ & & 0.001 & & & 0 & & \\ & & & 0.001 & & & & \\ & & & & 0.5 & & & \\ & & 0 & & & 0.5 & & \\ & & & & & & 0.001 & \\ & & & & & & & 0.001 \end{bmatrix}$$

$$R = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}$$

The obtained bandwidth is 5.5 Hz.

The reference values for all state variables are calculated based on the model of the flexible robot (chapter 5), given the desired angular position, velocity and acceleration of each link;

$$\begin{aligned} q_{j2d} &= -(A_{22}\ddot{\theta}_{l2d} + A_{23}\ddot{\theta}_{l3d} + B_2(\theta_{l2d}, \theta_{l3d}, \dot{\theta}_{l2d}, \dot{\theta}_{l3d}))/K_{j2} \\ q_{j3d} &= -(A_{23}\ddot{\theta}_{l2d} + I_{z3}\ddot{\theta}_{l3d} + B_3(\theta_{l2d}, \theta_{l3d}, \dot{\theta}_{l2d}, \dot{\theta}_{l3d}))/K_{j3} \\ \theta_{m2d} &= r_2(\theta_{l2d} - q_{j2d}) \\ \theta_{m3d} &= r_3(\theta_{l3d} - q_{j3d}) \end{aligned}$$

A smooth trajectory is generated based on a ninth order polynomial The movement of the tip is transformed to the angular position, velocity and acceleration of each link, by use of the inverse kinematic equations of a two link robot (appendix C).

The velocities are calculated by differentiation (chapter 5). A second order differentiation rule is used. To overcome the problem of sensitivity to high frequency noise of the differentiator, the measured signals are filtered with digital low pas filters. In addition the filters reduce the spillover problem. The used filters are second order butterworth filters with a cut–off frequency of 13 Hz. The filters are added in the model. This extends the model order with 4. Using higher order filters and higher order differentiation rules will improve the performance but can not be implemented because of the limitations in available computation power.

The state feedback gains of this extended model, are calculated using the pole placement approach where the closed loop poles are chosen as

- the closed loop poles obtained by applying optimal control theory to the robot model,
- the system poles that originate from the filter dynamics.

9.6 Experimental Results

The experimental results of the controller presented above, are compared to the performance of a rigid controller.
Four controllers are implemented:

1. A PID rigid controller in velocity command with appropriate feedforward.
2. A nonlinear decoupling rigid controller.
3. A nonlinear decoupling controller with feedforward and trajectory calculation based on the flexible model. Only rigid variables are fed back.
4. A nonlinear decoupling controller with feedforward and trajectory calculation based on the flexible model. Rigid and flexible coordinates are fed back.

All controllers are optimized (appropriate choice of feedback gains). Integration is added. The nonlinear Coulomb friction is compensated. The obtained bandwidth of controller 4 is 5.5 Hz. The bandwidth of

the nonlinear decoupling rigid controller (controller 2) is 1.6 Hz. For all controllers the same smooth trajectory is used. Tests have shown [68] [4] that using the smooth trajectory for the PID rigid controller gives the largest contribution to the improved dynamic performance (overshoot, oscillations, tracking error) compared to the performance of the standard industrial robot controller which is normally delivered with the robot. The tests described in this section will show that using a flexible controller instead of a classical PID controller gives also a significant improvement in terms of tracking accuracy. In addition the static error due to gravity is compensated.

In figure 9.7, the tip position trajectories of controller 4 (generalized nonlinear decoupling controller) and controller 1 (rigid controller in velocity command) are compared to the desired trajectory. The desired motion is a 0.254 m straight line, which makes an angle of 45 degrees with the horizontal plane. The motion has to be executed in 1.2 seconds. Figure 9.7 also compares the dynamic tip position error normal and tangential to the desired trajectory of the rigid controller and the flexible controller. The following improvements are obtained in comparison with rigid control:

- The maximum dynamic tracking error normal to the desired trajectory is reduced from 1.6 mm to 1.0 mm.
- The maximum dynamic tracking error tangential to the desired trajectory is reduced from 2.4 mm to 2.0 mm.
- The normal and tangential static errors due to gravitation are compensated.

In a second step the performance of the controller is evaluated by using the digitizing board. The difference is that the tip position is measured directly instead of being calculated from the extra encoder measurements. The motion is executed at low (in 3 seconds) and at high velocity (in 1.2 seconds). The end effector path measured at low velocity is considered as the reference path, with which the high velocity path is compared. The same conclusions can be made. The maximum deviation for controller 1 is 1.63 mm and for controller 4 0.8 mm. The static deflection is compensated. As in the identification measurements (section 9.4.2), the measured static end effector deflection is 25 % higher than the deflection calculated from the encoder measurements.

A more detailed analysis of the behaviour of the generalized nonlinear decoupling controller is obtained by comparing the tracking errors of the four implemented controllers. Figure 9.8 gives the tip position error normal and tangential to the desired trajectory of the two rigid controllers (controllers 1 and 2). Figure 9.9 compares the normal and tangential tracking errors of the full flexible controller and the tracking errors of the flexible controller without feedback of the flexible coordinates (controllers 3 and 4).
Using a nonlinear decoupling controller in torque command (controller 2) instead of a PID rigid conroller in velocity command (controller 1) gives no significant improvement. The mean value of the tracking error is reduced, but the maximum tracking error is of the same order of magnitude. To obtain better results it is necessary to have more accurate estimates of the model parameters and of the motor friction.
Comparison of the two flexible controllers (controllers 3 and 4) and the two rigid controllers (controllers 1 and 2) shows that the largest improvement is the compensation of the static deflection (the normal and tangential static deflections in the final position are 2.85 mm). But there is also a reduction of the dynamic tracking error.
Feeding back rigid and flexible coordinates (controller 4) instead of only rigid coordinates (controller 3) has no effect. In both cases there are no vibrations. Also for the rigid controllers (controllers 1 and 2) no vibrations are observed. The reason is the use of smooth trajectories.

From these experiments it can be concluded that some improvements can be obtained by implementing a flexible controller on an industrial robot. The improvements are small due to the limited flexibility of the robot and due to the difficulties in obtaining very accurate estimates of model parameters and motor friction torques.

9.7 Conclusion.

The nonlinear flexible controller is experimentally verified on a KUKA IR 161/60 industrial robot. Comparison with a controller designed for rigid robots shows that even for this robot with limited flexibility, the dynamic tracking error is reduced by a factor two, and in addition the static deflection is compensated.

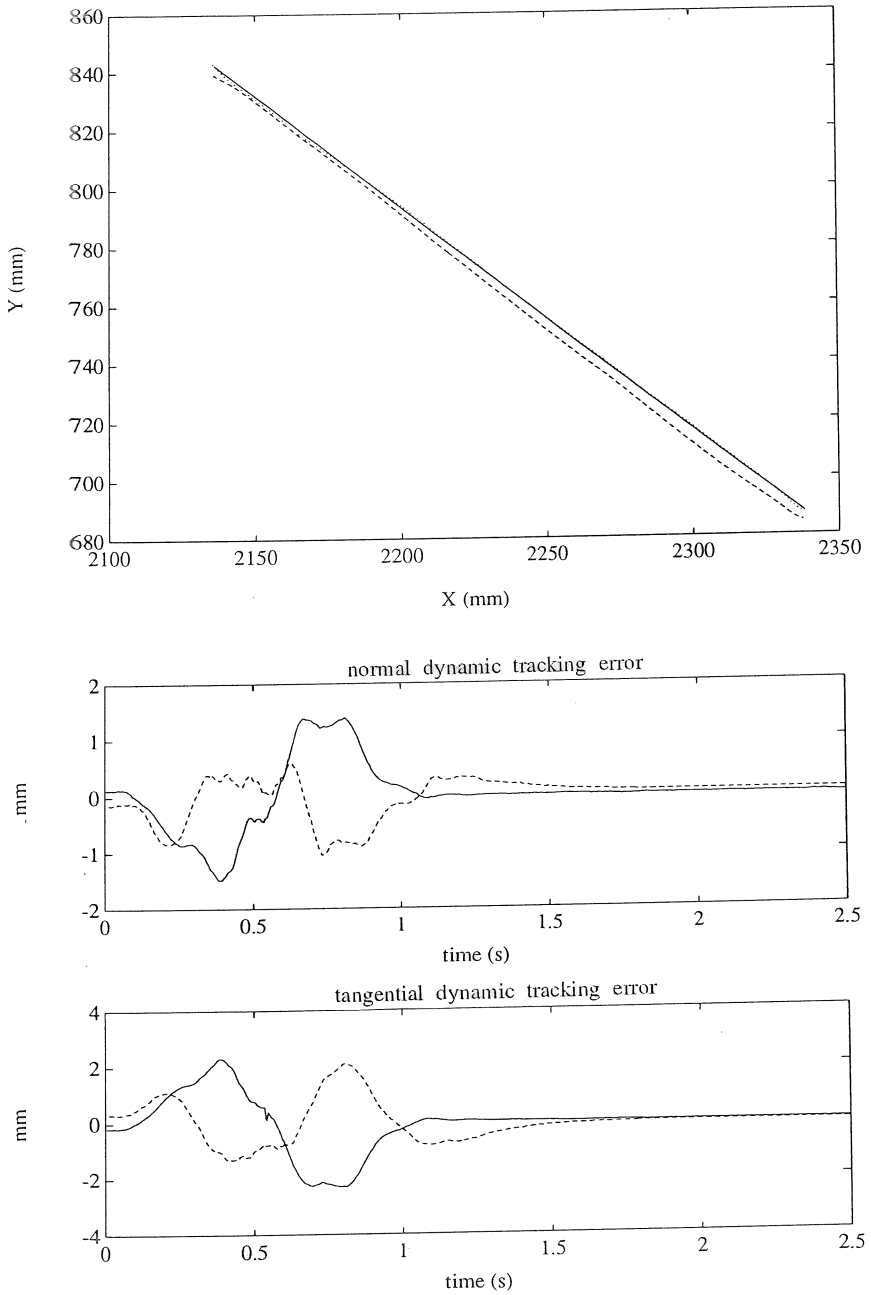

Figure 9.7: The tip position trajectory of a generalized nonlinear decoupling controller (dotted line) and a rigid controller (dashed line) compared with the desired trajectory (solid line). Bottom figures: dynamic tracking error (without static errors) normal and tangential to the desired trajectory in function of time (rigid controller: solid line, generalized nonlinear decoupling controller: dashed line).

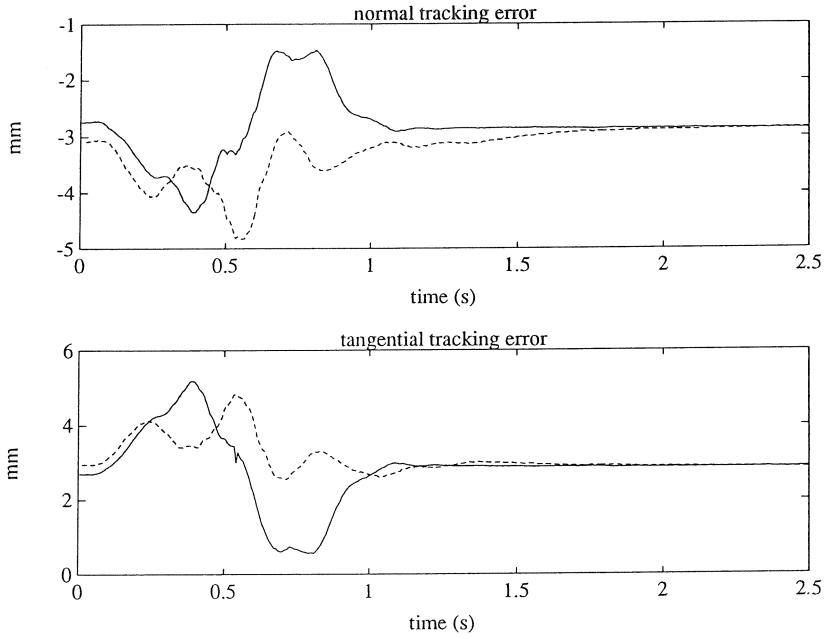

Figure 9.8: Tracking error normal and tangential to the desired trajectory in function of time for a rigid controller (velocity command: solid line, torque command: dashed line).

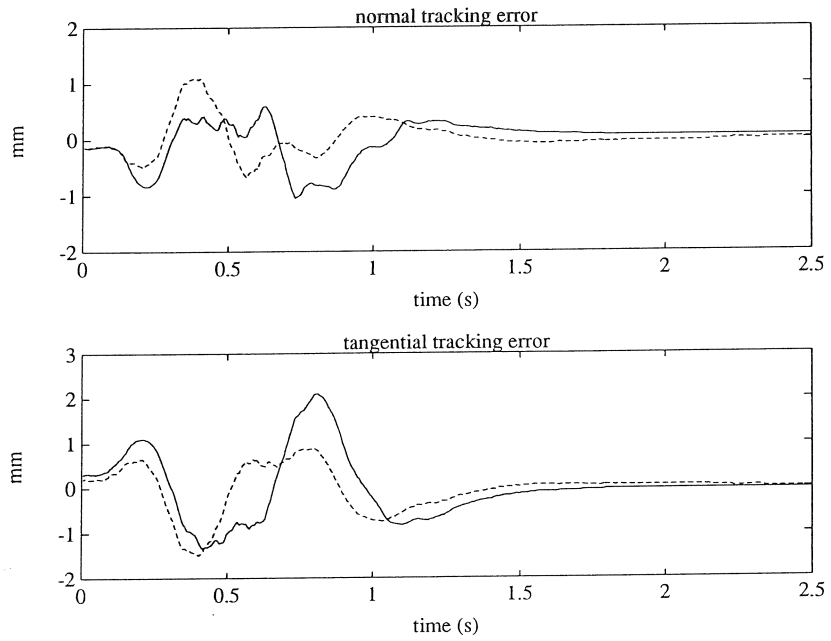

Figure 9.9: Tracking error normal and tangential to the desired trajectory in function of time for a generalized nonlinear decoupling controller (with feedback of flexible coordinates: solid line, without feedback of flexible coordinates: dashed line) .

Chapter 10

General conclusion

Flexibility of joints or links limits the achievable performance of traditional control designed for rigid robots. The aim of the research reported in this dissertation is to supplement traditional control , first to prevent or damp out undesired oscillations resulting from excitation of structural dynamics, and second to improve the tracking of a reference trajectory designed for a rigid robot. Therefore a new control algorithm is developed and verified experimentally on test set ups and on an industrial robot.

The developed control algorithm is based on an analytical model describing the dynamic behavior of the robot. To this end, a dynamic model of multi–link robotic manipulators with compliant joints and links is derived using the Lagrangian assumed modes formulation. The model is characterised by highly nonlinear equations with nonlinear couplings between the variables of motion. The final form of the equations is organized in a standard form similar to rigid manipulator equations.

The proposed control method consists of

- a nonlinear feedforward or feedback to cancel out nonlinear terms in the rigid body dynamics.

- linearization of the remaining nonlinearities, and linear feedback of all state variables (rigid and flexible coordinates). This ensures proper stability of the system about the desired state.

- compensation of static and dynamic deflection. This improves the tracking of a reference trajectory. The compensation is achieved by calculation of the reference values of all state variables based on the model of the flexible robot.

The algorithm is applicable to flexible joint robots as well as to flexible link robots. The compensation torque can be a nonlinear feedforward torque or a nonlinear feedback torque. Using the feedforward dynamics compensation may have some implementational advantages, therefore in the experiments feedforward compensation is applied.

The proposed control method is applied to a simulation model of a two–link flexible robot. The simulations show that, in comparison with a rigid controller, the excited vibrations are damped out very rapidly, and that tracking errors are reduced to negligible values.

Apart from the theoretical research of nonlinear control of flexible robots, an important contribution of this dissertation is the application of the proposed control algorithm to experimental robots. To this end, the problems associated with practical implementation are solved.
A first problem is the identification of the dynamic parameters of the models. Since the achievable performance of the control system is limited by the model accuracy, a lot of attention is paid to this identification. An overview of methods, which have proven to be useful in the identification of the different setups, is given. Practice has shown that splitting up the identification problem into several smaller subproblems gives more reliable parameter estimates than trying to identify all the parameters in one step. Using different identification techniques gives an idea of the accuracy of the obtained parameters.
Another problem associated with implementation on experimental robots is the choice of sensors and the estimation of the unmeasurable state variables. To estimate the unmeasurable state variables, a discrete nonlinear state estimator is proposed and implemented on the test set ups. However, for the proposed nonlinear control method and for the used sensors, the only states which can not be measured directly are the velocities. It is shown that calculation of velocities by differentiation also gives good results with less calculation time, provided that the measurements are filtered with digital low pass filters. Low pass filtering is also needed to reduce spillover problems.

The filter dynamics are taken into account in the control. Real test structures show significant nonlinear motor friction. Nonlinear friction has a large effect on the design of a controller. It causes limit cycles, steady state errors and tracking errors which can not be corrected with plain state feedback. Steady state errors are eliminated using integral feedback, and limit cycles are avoided by restriction of the closed loop bandwith. In addition, a strategy is proposed in this dissertation which reduces limit cycles by phasing out the feedback once the final position is reached. In addition to these techniques, which don't need identification of the nonlinear friction, a nonlinear motor friction compensation scheme is implemented on the experimental manipulators. To this end, the nonlinear friction is first identified.
A last problem associated with the implementation on an experimental robot are the computational capabilities of available microprocessors. The computational capabilities of the used measurement and control unit are not sufficient for high sample-rate control of continuous time controllers. Therefore discrete control design is used. Computation time is minimized by striving after simple and straightforward solutions. However, at the moment, newer processors with increased computational capabilities are available. Using these processors gives the possibility to choose higher sample–frequencies.

The proposed control algorithm is applied to experimental test set ups with increasing complexity and to an industrial robot.
First, identification and control is applied to a flexible one–link robot. Because of the linearity of the dynamic model, the problem reduces to a linear identification and control problem. The experiments show that the obtained controller allows accurate tracking and positioning with very limited overshoot, oscillation and tracking errors.
An experimental two link manipulator with a flexible first joint is then constructed and the proposed nonlinear control method is applied to this test set up. The experiments show the superior performance of the controller compared to classical rigid control: there are no vibrations and the tracking error and steady state error are reduced considerably.
Experimental results on a KUKA IR 161/60 industrial robot show that it is also possible to apply the nonlinear flexible controller to industrial robots. The flexibility of this industrial robot, designed to be very stiff and accurate, is limited. The main flexibilities are located in

the joints. The experiments show that even for this "stiff" industrial robot, in comparison with classical rigid control some improvements can be obtained: the dynamic tracking error is halved and the static deflection due to gravity is compensated.

The author of this dissertation hopes that this work will contribute to the improvement of the performance of traditional robots in which flexibility of drive systems remains an important problem. More important, the author hopes also that this work will contribute to the development of a new generation of fast and accurate manipulators, in which the use of lightweight arms, with increased structural flexibility, is required in order to achieve higher motion speeds, better energy efficiency, safer operation and improved mobility.

Appendix A

Nonlinear decoupling control of nonlinear systems

This appendix describes the nonlinear decoupling control theory developed by Freund [28] and gives as example the nonlinear decoupling control of a one–link flexible robot arm.

A.1 Nonlinear decoupling control theory

The following general nonlinear time–variable system is basic:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{A}(\mathbf{x},t) + \mathbf{B}(\mathbf{x},t)\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}(\mathbf{x},t) + \mathbf{D}(\mathbf{x},t)\mathbf{u}(t) \end{aligned} \tag{A.1}$$

where $\mathbf{x}(t)$ is the n–dimensional state vector; $\mathbf{u}(t)$ and $\mathbf{y}(t)$ are the m–dimensional input and output vectors respectively; and $\mathbf{A}(\mathbf{x},t)$, $\mathbf{B}(\mathbf{x},t)$, $\mathbf{C}(\mathbf{x},t)$ and $\mathbf{D}(\mathbf{x},t)$ are of compatible order. Related to system A.1, a nonlinear operator N_A^k is defined for k = 1, 2,...

$$\begin{aligned} N_A^k C_i(\mathbf{x},t) &= \tfrac{\delta}{\delta t}\, N_A^{k-1} C_i(\mathbf{x},t) \\ &+ \left[\tfrac{\delta}{\delta \mathbf{x}}\, N_A^{k-1} C_i(\mathbf{x},t) \right] \mathbf{A}(\mathbf{x},t) \end{aligned} \tag{A.2}$$

If C_i and $\mathbf{A}$ are independent of t, the operator contains only the second term on the right hand side of equation A.2. In equation A.2

$N_A^0 C_i(\mathbf{x},t) = C_i(\mathbf{x},t)$ and $C_i(\mathbf{x},t)$ is the ith component of $\mathbf{C}(\mathbf{x},t)$ (i= 1, 2, ...,m).

The differential order, d_i, of the nonlinear system can be defined as follows (with $\mathbf{D}_i(\mathbf{x},t)$ the ith row vector of $\mathbf{D}(\mathbf{x},t)$):

$$\begin{aligned} d_i &= 0, && for\ \mathbf{D}_i(\mathbf{x},t) \neq 0, \\ d_i &= min\{\, j : \left[\tfrac{\delta}{\delta \mathbf{x}} N_A^{j-1} \mathbf{C}_i(\mathbf{x},t)\right] \mathbf{B}(\mathbf{x},t) \neq 0, && j = 1,2,...,n\,\}, \\ & && for\ \mathbf{D}_i(\mathbf{x},t) = 0 \end{aligned} \tag{A.3}$$

where it is assumed that d_i (i=1, 2,...,m) remains constant for all $\mathbf{x}$ and t in the considered range.

The goal is to find a feedback

$$\mathbf{u}(t) = \mathbf{F}(\mathbf{x},t) + \mathbf{G}(\mathbf{x},t)\ \mathbf{w}(t) \tag{A.4}$$

with $\mathbf{w}$ as new m–dimensional reference input vector and $\mathbf{F}$(x,t) and $\mathbf{G}$(x,t) of compatible order, such that the overall system (from equation A.4 in application to equation A.1) is decoupled and has arbitrary designated poles. This can be done by a feedback law of the type

$$\mathbf{F}(\mathbf{x},t) = \mathbf{F}^*(\mathbf{x},t) = \mathbf{F}_1^*(\mathbf{x},t) + \mathbf{F}_2^*(\mathbf{x},t) \tag{A.5}$$

where

$$\mathbf{F}_1^*(\mathbf{x},t) = -\mathbf{D}^{*-1}(\mathbf{x},t)\ \mathbf{C}^*(\mathbf{x},t) \tag{A.6}$$

$$\mathbf{F}_2^*(\mathbf{x},t) = -\mathbf{D}^{*-1}(\mathbf{x},t)\ \mathbf{M}^*(\mathbf{x},t) \tag{A.7}$$

and

$$\mathbf{G}(\mathbf{x},t) = \mathbf{G}^*(\mathbf{x},t) = \mathbf{D}^{*-1}(\mathbf{x},t)\ \Lambda \tag{A.8}$$

$\mathbf{F}_1^*(\mathbf{x},t)$ represents the part of the feedback that yields decoupling, while $\mathbf{F}_2^*(\mathbf{x},t)$ performs the control part with arbitrary pole assignment; the input gain of the decoupled branches can be chosen by $\mathbf{G}$(x,t). The matrices in equations A.6, A.7 and A.8 are given as follows:

$\mathbf{D}^*(\mathbf{x},t)$ is a mxm matrix with the ith row (i=1,2,...,m)

$$\begin{aligned} \mathbf{D}_i^*(\mathbf{x},t) &= \mathbf{D}_i(\mathbf{x},t) && for\ d_i = 0 \\ \mathbf{D}_i^*(\mathbf{x},t) &= \left[\tfrac{\delta}{\delta \mathbf{x}}\ N_A^{d_i-1} C_i(\mathbf{x},t)\right] \mathbf{B}(\mathbf{x},t) && for\ d_i \neq 0 \end{aligned} \tag{A.9}$$

Here it is assumed that the rank of $\mathbf{D}^*$ remains constant for all $\mathbf{x}$ and t in the considered range.
$\mathbf{C}^*(\mathbf{x},t)$ is a m–dimensional vector with the ith component

$$C_i^*(\mathbf{x},t) = N_A^{d_i} C_i(\mathbf{x},t) \tag{A.10}$$

$\mathbf{M}^*(\mathbf{x},t)$ is a m–dimensional vector whose ith component is

$$\begin{array}{ll} M_i^*(\mathbf{x},t) = 0 & for \;\; d_i = 0 \\ M_i^*(\mathbf{x},t) = \sum_{k=0}^{d_i-1} \alpha_{k,i}\, N_A^k C_i(\mathbf{x},t) & for \;\; d_i \neq 0 \end{array} \tag{A.11}$$

The mxm matrix $\mathit{\Lambda}$ is a diagonal matrix; the elements are λ_i for i=1, 2, .., m.

The fundamental equation for the derivation of this general control law is the representation of systems (equation A.1) in terms of

$$\mathbf{y}^*(t) = \mathbf{C}^*(\mathbf{x},t) + \mathbf{D}^*(\mathbf{x},t)\,\mathbf{u}(t) \tag{A.12}$$

with $\mathbf{C}^*$ and $\mathbf{D}^*$ as given in equations A.10 and A.9 and $\mathbf{y}^*$ a m–dimensional vector with the ith component $y_i^{d_i}$. The ith component of equation A.12 has the form

$$y_i^{d_i}(t) = C_i^*(\mathbf{x},t) + \mathbf{D}_i^*(\mathbf{x},t)\,\mathbf{u}(t) \qquad i = 1,2,...,m \tag{A.13}$$

Application of the feedback law A.4 to system A.1 with $\mathbf{F}^*(\mathbf{x},t)$ and $\mathbf{G}^*(\mathbf{x},t)$ yields the overall behaviour of all input/output pairs $y_i(t)$, $w_i(t)$ (i=1, 2,...,m):

$$y_i^{d_i}(t) + \alpha_{d_{i-1},i}\, y_i^{d_{i-1}}(t) + ... + \alpha_{0,i}\, y_i(t) = \lambda_i\, w_i(t) \tag{A.14}$$

where the $\alpha_{k,i}$ and λ_i can be chosen arbitrarily, and w_i has the same physical dimension as y_i.

A.2 Example: nonlinear decoupling control of a one–link flexible robot arm

The model of a one–link flexible robot arm is derived in chapter 3 (equation 3.7). The model can be written in the general form (equation A.1)

$$\begin{array}{l} \dot{\mathbf{x}}(t) = \mathbf{A}(\mathbf{x},t) + \mathbf{B}(\mathbf{x},t)\mathbf{u}(t) \\ \mathbf{y}(t) = \mathbf{C}(\mathbf{x},t) + \mathbf{D}(\mathbf{x},t)\mathbf{u}(t) \end{array} \tag{A.15}$$

with

$$\begin{aligned}\mathbf{x} &= [\alpha \;\; \dot{\alpha} \;\; q_1 \;\; \dot{q}_1]^T \\ &= [x_1 \;\; x_2 \;\; x_3 \;\; x_4]^T\end{aligned}$$

$$\mathbf{A}(\mathbf{x},t) = \mathbf{A}(\mathbf{x}) = \begin{bmatrix} x_2 \\ 0 \\ x_4 \\ -\omega_1^2 x_3 - 2\zeta_1\omega_1 x_4 \end{bmatrix}$$

$$\mathbf{B}(\mathbf{x},t) = \mathbf{B} = \frac{1}{I_t}\begin{bmatrix} 0 \\ 1 \\ 0 \\ \frac{d\Psi_1}{dx''}(0) \end{bmatrix}$$

where
$I_t = I_b + I_h + I_p + m_p l^2$ is the total inertia of beam, hub and payload,
l is the length of the beam,
$\Psi_1(x)$ is the constrained shape function of the first flexible mode,
$q_1(t)$ is the generalized coordinate of the first flexible mode,
$\alpha(t)$ is the rigid body rotation angle,
ω_1 is the first cantilevered mode eigenpulsation and
ζ_1 is the modal damping of the first flexible mode.
Only one flexible mode is considered.
The tip rotation

$$y_e = l\alpha + \Psi_1(l)q_1 \tag{A.16}$$

is chosen as output. Hence,

$$\mathbf{C}(\mathbf{x},t) = \mathbf{C}(\mathbf{x}) = l\,x_1 + \Psi_1(l)\,x_3$$

$$\mathbf{D}(\mathbf{x},t) = 0$$

Calculation of $N_A^0 C(\mathbf{x})$, $N_A^1 C(\mathbf{x})$ and $N_A^2 C(\mathbf{x})$:

$$
\begin{aligned}
N_A^0 C(\mathbf{x}) &= C(\mathbf{x}) \\
N_A^1 C(\mathbf{x}) &= \left[\tfrac{\delta}{\delta \mathbf{x}} C(\mathbf{x}) \right] \mathbf{A}(\mathbf{x}) \\
&= [\ l \quad 0 \quad \Psi_1(l) \quad 0\ ]\, \mathbf{A}(\mathbf{x}) \\
&= l\, x_2 + \Psi_1(l)\, x_4 \\
N_A^2 C(\mathbf{x}) &= \left[\tfrac{\delta}{\delta \mathbf{x}} N_A^1 C(\mathbf{x}) \right] \mathbf{A}(\mathbf{x}) \\
&= [\ 0 \quad l \quad 0 \quad \Psi_1(l)\ ]\ \mathbf{A}(\mathbf{x}) \\
&= -\Psi_1(l)\, \omega_1^2\, x_3 - \Psi_1(l)\, 2\zeta_1 \omega_1\, x_4
\end{aligned}
$$

Determination of the differential order:

Since $\mathbf{D}(\mathbf{x}, t) = 0$, from equation A.3,

$$
d_i = min\{\ j : \left[\frac{\delta}{\delta \mathbf{x}} N_A^{j-1} \mathbf{C}_i(\mathbf{x}, t) \right] \mathbf{B}(\mathbf{x}, t) \neq 0, \qquad j = 1, 2, ..., 4\ \}
$$

for j=1, $\left[\frac{\delta}{\delta \mathbf{x}} N_A^0 \mathbf{C}(\mathbf{x}) \right] \mathbf{B} = [\ l \quad 0 \quad \Psi_1(l) \quad 0\ ]\ \mathbf{B} = 0$

for j=2, $\left[\frac{\delta}{\delta \mathbf{x}} N_A^1 \mathbf{C}(\mathbf{x}) \right] \mathbf{B} = [\ 0 \quad l \quad 0 \quad \Psi_1(l)\ ]\ \mathbf{B} = \frac{l + \Psi_1 \frac{d\Psi_1}{dx}(0)}{I_T} \neq 0$

The differential order d is thus equal to 2.

Calculation of the feedback law:

The feedback law is given by (equation A.4, A.5, A.6, A.7 and A.8)

$$
\begin{aligned}
\mathbf{u}(t) &= \mathbf{F}(\mathbf{x}, t) + \mathbf{G}(\mathbf{x}, t)\, \mathbf{w}(t) \\
&= -\mathbf{D}^{*-1}(\mathbf{x}, t)\ (\ \mathbf{C}^*(\mathbf{x}, t) + \mathbf{M}^*(\mathbf{x}, t)\ ) + \mathbf{D}^{*-1}(\mathbf{x}, t)\ \Lambda\ \mathbf{w}(t)
\end{aligned}
\tag{A.17}
$$

Since $d \neq 0$, from equation A.9, A.10 and A.11

$$\begin{aligned} \mathbf{D}^*(\mathbf{x},t) &= \left[\frac{\delta}{\delta \mathbf{x}} N_A^1 C(\mathbf{x},t) \right] \mathbf{B}(\mathbf{x},t) \\ &= \left[\begin{array}{cccc} 0 & l & 0 & \Psi_1(l) \end{array} \right] \mathbf{B} \\ &= \frac{l + \Psi_1 \frac{d\Psi_1}{dx}^T(0)}{I_T} \end{aligned}$$

$$\begin{aligned} C^*(\mathbf{x},t) &= N_A^2 C(\mathbf{x},t) \\ &= -\Psi_1(l)\, \omega_1^2\, x_3 - \Psi_1(l)\, 2\zeta_1 \omega_1\, x_4 \end{aligned}$$

$$\begin{aligned} M^*(\mathbf{x},t) &= \alpha_0\, N_A^0 C(\mathbf{x}) + \alpha_1\, N_A^1 C(\mathbf{x}) \\ &= \alpha_0\, (\, l x_1 + \Psi_1(l)\, x_3\,) + \alpha_1\, (\, l x_2 + \Psi_1(l)\, x_4\,) \end{aligned}$$

Application of the feedback law A.17 to system A.15 yields the overall behaviour of input/output pair $y_e(t)$, $w(t)$

$$\ddot{y}_e(t) + \alpha_1\, \dot{y}_e(t) + \alpha_0\, y_e(t) = \lambda\, w(t) \qquad \text{(A.18)}$$

By chosing α_1, α_0 and λ appropriately, the desired dynamic behaviour of the output y_e is obtained.

Behaviour of the controlled system:

By applying the nonlinear decoupling technique, for each inputvariable, a linear subsystem can be separated. The order of the separated subsustem is equal to the differential order.

For the one–link flexible robot the differential order is 2. There is one inputvariable, hence, one linear subsystem of order 2 can be separated.

Consider the transferfunction relating the reference input w to the flexible coordinate q_1. Application of the feedback law A.17 to system A.15 yields

$$\frac{q_1}{w}(s) = \frac{\lambda\, s^2}{(s^2 + \alpha_1\, s + \alpha_0)(s^2 + \frac{2\zeta_1 \omega_1 l}{l + \Psi_1 \frac{d\Psi_1}{dx}^T(0)} s + \frac{\omega_1^2 l}{l + \Psi_1 \frac{d\Psi_1}{dx}^T(0)})}$$

The parameters α_1 and α_0 of the first part of the denominator $s^2 + \alpha_1\, s + \alpha_0$ are chosen to become the desired dynamic behaviour for the input/output pair $y_e(t)$, $w(t)$ (equation A.18).
The second part of the denominator

$$s^2 + \frac{2\zeta_1\omega_1 l}{l+\Psi_1 \frac{d\Psi_1}{dx}\Big|_{x=l}(0)} s + \frac{\omega_1^2 l}{l+\Psi_1 \frac{d\Psi_1}{dx}\Big|_{x=l}(0)}$$

remains uncontrolled and may cause undesired oscillations.

Appendix B

Computed torque method, singular perturbation method and generalized nonlinear decoupling method applied to flexible joint robots

In this appendix three different control methods are applied to flexible joint robots. Section B.2 derives the computed torque solution. Section B.3 derives the generalized nonlinear decoupling solution. Section B.4 derives singular perturbation solution. In section B.5 the three solutions are compared. For the three methods the same feedforward compensation torque is obtained, if the feedforward compensation scheme is applied. The methods differ in the choice of feedback variables.

B.1 Model of a flexible joint robot

The dynamic model of a flexible joint robot is given by:

$$\mathbf{M}_l(\boldsymbol{\theta}_l)\,\ddot{\boldsymbol{\theta}}_l + \mathbf{F}_r(\boldsymbol{\theta}_l, \dot{\boldsymbol{\theta}}_l) + \mathbf{K}_j\,\mathbf{q}_j = 0 \qquad \text{(B.1)}$$

$$\mathbf{J}\,\ddot{\theta}_m - \mathbf{K}_j\,\mathbf{q}_j = \mathbf{T} \tag{B.2}$$

where, for a manipulator with N rigid degrees of freedom,

θ_l	$= [\theta_{l1}\ \theta_{l2}\ ...\theta_{lN}]^T$ is the vector of link positions (joint angles after the elastic transmission),
θ_m	$= [\theta_{m1}\ \theta_{m2}\ ...\theta_{mN}]^T$ is the vector of motor positions,
$\mathbf{q}_j$	$= \theta_l - \theta_m = [q_{j1}\ q_{j2}\ ...q_{jN}]^T$ is the vector of generalized coordinates associated with the joint flexibility
$\mathbf{T}$	is the vector of motor torques,
$\mathbf{J}$	$= Diag\{I_{mi}\}$, with I_{mi} the rotor/gear assembly inertia of motor i,
$\mathbf{M}_l(\theta_l)$	$= \mathbf{M}_r(\theta_l) - \mathbf{J}$, with $\mathbf{M}_r(\theta_l)$ the generalized inertia matrix (the inertia of links is separated from the rotor/gear assembly inertia)
$\mathbf{F}_r(\theta_l, \dot{\theta}_l)$	contains centrifugal, coriolis and gravitational forces
$\mathbf{K}_j$	$= Diag\{K_{ji}\}$ with K_{ji} the spring constant of joint i,

In this model, elastic mechanical coupling between a joint and link is modeled as a torsional spring: the relative motion between the joint rotor and elastically coupled link, $(\theta_{li} - \theta_{mi}) = q_{ji}$, gives a force $K_{ji}q_{ji}$

B.2 Computed torque method

The computed torque method consists of finding a coordinate transformation

$$\mathbf{y} = \mathbf{F}(\theta_l, \mathbf{q}_j)$$

and a nonlineaer feedback

$$\mathbf{T} = \mathbf{a}(\theta_l, \mathbf{q}_j) + \mathbf{B}(\theta_l, \mathbf{q}_j)\ \mathbf{v}$$

such that the transformed state y satisfies the linear system

$$\dot{\mathbf{y}} = \mathbf{A}_l\ \mathbf{y} + \mathbf{B}_l\ \mathbf{v}$$

The computed torque solution for flexible joint robots, found by Bortoff and Spong [9], defines the transformed state

$\mathbf{y} = [\ \mathbf{y}_1\ \mathbf{y}_2\ \mathbf{y}_3\ \mathbf{y}_4\ ]^T$ as

$$\mathbf{y}_1 = \theta_l \tag{B.3}$$

$$\mathbf{y}_2 = \dot{\theta}_l \tag{B.4}$$

$$\mathbf{y}_3 = \ddot{\theta}_l = -\mathbf{M}_l^{-1}(\ \mathbf{F}_r + \mathbf{K}_j\mathbf{q}_j) \tag{B.5}$$

$$\mathbf{y}_4 = \theta_l^{III} = -\dot{\mathbf{M}}_l^{-1}(\ \mathbf{F}_r + \mathbf{K}_j\mathbf{q}_j) - \mathbf{M}_l^{-1}(\ \dot{\mathbf{F}}_r + \mathbf{K}\dot{\mathbf{q}}_j) \tag{B.6}$$

The transformed state satisfies the linear system

$$\begin{bmatrix} \dot{\mathbf{y}}_1 \\ \dot{\mathbf{y}}_2 \\ \dot{\mathbf{y}}_3 \\ \dot{\mathbf{y}}_4 \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \mathbf{y}_3 \\ \mathbf{y}_4 \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{I} \end{bmatrix} \mathbf{v}$$

with

$$\mathbf{v} = \dot{\mathbf{y}}_4 = \theta_l^{IV}$$

Differentiating B.6 gives

$$\begin{aligned} \mathbf{v} &= -\ddot{\mathbf{M}}_l^{-1}(\ \mathbf{F}_r + \mathbf{K}_j\mathbf{q}_j) - 2\dot{\mathbf{M}}_l^{-1}(\ \dot{\mathbf{F}}_r + \mathbf{K}_j\dot{\mathbf{q}}_j) - \mathbf{M}_l^{-1}(\ \ddot{\mathbf{F}}_r + \mathbf{K}_j\ddot{\mathbf{q}}_j) \\ &= -\ddot{\mathbf{M}}_l^{-1}(\ \mathbf{F}_r + \mathbf{K}_j\mathbf{q}_j) - 2\dot{\mathbf{M}}_l^{-1}(\ \dot{\mathbf{F}}_r + \mathbf{K}_j\dot{\mathbf{q}}_j) \\ &\quad -\mathbf{M}_l^{-1}(\ \ddot{\mathbf{F}}_r - \mathbf{K}_j\mathbf{M}_l^{-1}[\mathbf{F}_r + \mathbf{K}_j\mathbf{q}_j] - \mathbf{K}_j\mathbf{J}^{-1}\mathbf{K}_j\mathbf{q}_j) \\ &\quad +\mathbf{M}_l^{-1}\mathbf{K}_j\mathbf{J}^{-1}\mathbf{T} \\ &= \mathbf{F}(\theta_l, \dot{\theta}_l, \ddot{\theta}_l, \theta_l^{III}, \mathbf{q}_j, \dot{\mathbf{q}}_j) + \mathbf{M}_l^{-1}\mathbf{K}_j\mathbf{J}^{-1}\mathbf{T} \end{aligned}$$

The nonlinear feedback is given by

$$\mathbf{T} = -\mathbf{J}\mathbf{K}_j^{-1}\mathbf{M}_l(\theta_l)\ \mathbf{F}(\theta_l, \dot{\theta}_l, \ddot{\theta}_l, \theta_l^{III}, \mathbf{q}_j, \dot{\mathbf{q}}_j) + \mathbf{J}\mathbf{K}_j^{-1}\mathbf{M}_l(\theta_l)\ \mathbf{v} \tag{B.7}$$

Given an input trajectory of position, velocity, acceleration, jerk and derivative for each joint (θ_{ld}, $\dot{\theta}_{ld}$, $\ddot{\theta}_{ld}$, θ_{ld}^{III}, θ_{ld}^{IV}) , the input to the linear system, $\mathbf{v}$, is computed as a linear combination of the errors,

$$\mathbf{v} = \theta_{ld}^{IV} + \mathbf{K}_j(\theta_{ld}^{III} - \theta_l^{III}) + \mathbf{K}_a(\ddot{\theta}_{ld} - \ddot{\theta}_l) + \mathbf{K}_v(\dot{\theta}_{ld} - \dot{\theta}_l) + \mathbf{K}_p(\theta_{ld} - \theta_l)$$

In practice the time dependency of the mass matrix $\mathbf{M}_l$ can be neglected. The function $\mathbf{F}(\theta_l, \dot{\theta}_l, \ddot{\theta}_l, \theta_l^{III}, \mathbf{q}_j, \dot{\mathbf{q}}_j)$ in equation B.7 becomes

$$\begin{aligned} \mathbf{F}(\theta_l, \dot{\theta}_l, \ddot{\theta}_l, \theta_l^{III}, \mathbf{q}_j) &= -\mathbf{M}_l^{-1}(\ \ddot{\mathbf{F}}_r - \mathbf{K}_j\mathbf{M}_l^{-1}[\mathbf{F}_r + \mathbf{K}_j\mathbf{q}_j] \\ &\quad -\mathbf{K}_j\mathbf{J}^{-1}\mathbf{K}_j\mathbf{q}_j) \end{aligned}$$

If the computed torque solution is expressed as a feedforward compensation scheme, the control torque becomes

$$\mathbf{T} = \mathbf{T}_{ff} + \mathbf{J}\mathbf{K}_j^{-1}\mathbf{M}_l\ \mathbf{T}_{fb} \tag{B.8}$$

with

$$\mathbf{T}_{fb} = \mathbf{K}_j(\theta_{ld}^{III} - \theta_l^{III}) + \mathbf{K}_a(\ddot{\theta}_{ld} - \ddot{\theta}_l) + \mathbf{K}_v(\dot{\theta}_{ld} - \dot{\theta}_l) + \mathbf{K}_p(\theta_{ld} - \theta_l) \tag{B.9}$$

$$\begin{aligned}\mathbf{T}_{ff} \;=\; & \mathbf{J}\mathbf{K}_j^{-1}\ \ddot{\mathbf{F}}_r(\theta_{ld}, \dot{\theta}_{ld}) - \mathbf{J}\mathbf{M}_l^{-1}(\theta_{ld})\ [\mathbf{F}_r(\theta_{ld}, \dot{\theta}_{ld}) + \mathbf{K}_j\mathbf{q}_{jd}] \\ & -\mathbf{K}_j\mathbf{q}_{jd} + \mathbf{J}\mathbf{K}_j^{-1}\mathbf{M}_l(\theta_{ld})\ \theta_{ld}^{IV}\end{aligned} \tag{B.10}$$

The feedforward torque, $\mathbf{T}_{ff}$, can be simplified given

$$\begin{aligned}\ddot{\theta}_{ld} &= -\mathbf{M}_l^{-1}[\mathbf{F}_r(\theta_{ld}, \dot{\theta}_{ld}) + \mathbf{K}_j\mathbf{q}_{jd}] \\ \theta_{ld}^{IV} &= -\mathbf{M}_l^{-1}[\ddot{\mathbf{F}}_r(\theta_{ld}, \dot{\theta}_{ld}) + \mathbf{K}_j\ddot{q}_{jd}], \\ \mathbf{T}_{ff} &= \mathbf{J}\mathbf{K}_j^{-1}\ \ddot{\mathbf{F}}_r(\theta_{ld}, \dot{\theta}_{ld}) + \mathbf{J}\ddot{\theta}_{ld} - \mathbf{K}_j\mathbf{q}_{jd} \\ & -\mathbf{J}\mathbf{K}_j^{-1}[\ddot{\mathbf{F}}_r(\theta_{ld}, \dot{\theta}_{ld}) + \mathbf{K}_j\ddot{q}_{jd}] \\ & = \mathbf{J}\ddot{\theta}_{md} - \mathbf{K}_j\ \mathbf{q}_{jd}\end{aligned} \tag{B.11}$$

B.3 Generalized nonlinear decoupling method

The dynamic model (equations B.1 and B.2) can be written as:

$$\begin{bmatrix} \mathbf{M}_l + \mathbf{J} & \mathbf{M}_l \\ \mathbf{M}_l & \mathbf{M}_l \end{bmatrix} \begin{bmatrix} \ddot{\theta}_m \\ \ddot{\mathbf{q}}_j \end{bmatrix} + \begin{bmatrix} \mathbf{F}_r \\ \mathbf{F}_r + \mathbf{K}_j\mathbf{q}_j \end{bmatrix} = \begin{bmatrix} \mathbf{T} \\ \mathbf{0} \end{bmatrix} \tag{B.12}$$

The inverse of the mass matrix

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_l + \mathbf{J} & \mathbf{M}_l \\ \mathbf{M}_l & \mathbf{M}_l \end{bmatrix}$$

is

$$\mathbf{W} = \begin{bmatrix} \mathbf{J}^{-1} & -\mathbf{J}^{-1} \\ -\mathbf{J}^{-1} & \mathbf{J}^{-1} + \mathbf{M}_l^{-1} \end{bmatrix} = \begin{bmatrix} \mathbf{W}_{11} & \mathbf{W}_{12} \\ \mathbf{W}_{12}^T & \mathbf{W}_{22} \end{bmatrix}$$

The control law is split up into two feedback loops.
First loop: the inner loop (non linear) which realizes feedback linearization of the state variables containing the rigid part:

$$\begin{aligned}\mathbf{T} &= \mathbf{F}_r(\theta_l, \dot{\theta}_l) + \mathbf{W}_{11}^{-1}\mathbf{W}_{12}(\theta_l)\ (\mathbf{F}_r(\theta_l, \dot{\theta}_l) + \mathbf{K}_j\mathbf{q}_j) + \mathbf{W}_{11}^{-1}(\theta_l)\mathbf{T}_1 \\ &= -\mathbf{K}_j\ \mathbf{q}_j + \mathbf{J}\ \mathbf{T}_1\end{aligned} \tag{B.13}$$

Then, equation B.12 becomes:

$$\begin{aligned} \ddot{\theta}_m &= \mathbf{T}_1 \qquad (linear) \\ \ddot{\mathbf{q}}_j &= (\mathbf{W}_{12}^T \mathbf{W}_{11}^{-1} \mathbf{W}_{12} - \mathbf{W}_{22})(\theta_l) \, [\mathbf{F}_r(\theta_l, \dot{\theta}_l) + \mathbf{K}_j \mathbf{q}_j] + \\ &\quad \mathbf{W}_{12}^T \mathbf{W}_{11}^{-1}(\theta_l) \mathbf{T}_1 \\ &= \ddot{\theta}_l - \mathbf{T}_1 \end{aligned} \tag{B.14}$$

Second loop: the outer loop (linear), which ensures proper stability of the system about desired state $\mathbf{x}_d$, by linear feedback of the full state $\mathbf{x}$:

$$\mathbf{T}_1 = \ddot{\theta}_{md} - \mathbf{K}(\mathbf{x} - \mathbf{x}_d) \tag{B.15}$$

with $\mathbf{x} = [\theta_m \quad \mathbf{q}_j \quad \dot{\theta}_m \quad \dot{\mathbf{q}}_j \,]$ and with the feedback gain matrix $\mathbf{K}$ calculated from the partially linearized model B.14, linearized about the desired trajectory.

The complete feedback control law is obtained from equations B.13 and B.15 as

$$\mathbf{T} = -\mathbf{K}_j \mathbf{q}_j + \mathbf{J} \ddot{\theta}_{md} - \mathbf{J}\mathbf{K}(\mathbf{x} - \mathbf{x}_d) \tag{B.16}$$

If the generalized nonlinear decoupling method is expressed as a feedforward compensation scheme, the control law becomes

$$\mathbf{T} = \mathbf{T}_{ff}(\mathbf{x}_d) + \mathbf{J} \, \mathbf{T}_{fb} \tag{B.17}$$

with

$$\begin{aligned} \mathbf{T}_{ff}(\mathbf{x}_d) &= -\mathbf{K}_j \mathbf{q}_{jd} + \mathbf{J} \ddot{\theta}_{md} \\ \mathbf{T}_{fb} &= -\mathbf{K} \, (\mathbf{x} - \mathbf{x}_d) \end{aligned}$$

B.4 Singular Perturbation Method

The singular perturbation solution for flexible joint robots is formulated by Spong [65].

The model of the flexible joint manipulator is reformulated as a singular perturbation model given by

$$\ddot{\theta}_l = \mathbf{M}_l^{-1} \mathbf{z} - \mathbf{M}_l^{-1} \mathbf{F}_r(\theta_l, \dot{\theta}_l) \tag{B.18}$$

$$\boldsymbol{\mu} \ddot{\mathbf{z}} = -(\mathbf{J}^{-1} + \mathbf{M}_l^{-1}) \mathbf{z} + \mathbf{M}_l^{-1} \mathbf{F}_r(\theta_l, \dot{\theta}_l) + \mathbf{J}^{-1} u \tag{B.19}$$

where $\mathbf{z} = -\mathbf{K}_j(\theta_l - \theta_m) = -\mathbf{K}_j\mathbf{q}_j$
$\boldsymbol{\mu} = \mathbf{K}_j^{-1}$
$\mathbf{u} = \mathbf{T}$

The overall control strategy consists of a "rigid control", $\mathbf{u}_0$, designed for the rigid system

$$(\mathbf{M}_l + \mathbf{J})\,\ddot{\theta}_l + \mathbf{F}_r(\theta_l, \dot{\theta}_l) = \mathbf{u} \tag{B.20}$$

and a "corrective control", $\mathbf{u}_1$ to compensate for deviations of the flexible system response from the response of the rigid system:

$$\mathbf{u} = \mathbf{u}_0 + \boldsymbol{\mu}\,\mathbf{u}_1$$

The rigid control $\mathbf{u}_0$ is obtained using the computed torque method for the rigid system B.20

$$\mathbf{u}_0 = (\mathbf{M}_l + \mathbf{J})\,\mathbf{v} + \mathbf{F}_r(\theta_l, \dot{\theta}_l)$$

with

$$\mathbf{v} = \ddot{\theta}_{ld} + \mathbf{K}_v(\dot{\theta}_{ld} - \dot{\theta}_l) + \mathbf{K}_p(\theta_{ld} - \theta_l)$$

The corrective control $\mathbf{u}_1$ is based on the "reduced–order flexible model".
The reduced–order flexible model is formed by replacing $\mathbf{z}$ by $\mathbf{h}(\theta_l, \dot{\theta}_l, \mathbf{u}, \boldsymbol{\mu})$ (integral manifold approach) in B.18 and B.19.
Expanding the function $\mathbf{h}$ in terms of $\boldsymbol{\mu}$ as follows:

$$\mathbf{h}(\theta_l, \dot{\theta}_l, \mathbf{u}, \boldsymbol{\mu}) = \mathbf{h}_0 + \boldsymbol{\mu}\mathbf{h}_1 + \boldsymbol{\mu}^2\mathbf{h}_2 + \ldots$$

and substituting it in B.18 gives the reduced–order flexible model

$$\ddot{\theta}_l = -\mathbf{M}_l^{-1}\mathbf{F}_r + \mathbf{M}_l^{-1}(\mathbf{h}_0 + \boldsymbol{\mu}\mathbf{h}_1) + \mathcal{O}\,(\boldsymbol{\mu}^2) \tag{B.21}$$

Substituting $\mathbf{z} = \mathbf{h}(\theta_l, \dot{\theta}_l, \mathbf{u}, \boldsymbol{\mu})$ in B.19 yields the manifold condition

$$\boldsymbol{\mu}\ddot{\mathbf{h}} = -(\mathbf{J}^{-1} + \mathbf{M}_l^{-1})\mathbf{h} + \mathbf{M}_l^{-1}\mathbf{F}_r(\theta_l, \dot{\theta}_l) + \mathbf{J}^{-1}\mathbf{u} \tag{B.22}$$

or

$$\begin{aligned}\mu(\ddot{\mathbf{h}}_0 + \mu\ddot{\mathbf{h}}_1 + ...) = \; & -(\mathbf{J}^{-1} + \mathbf{M}_l^{-1})(\mathbf{h}_0 + \mu\mathbf{h}_1 + ...) \\ & +\mathbf{M}_l^{-1}\mathbf{F}_r(\theta_l, \dot{\theta}_l) + \mathbf{J}^{-1}(\mathbf{u}_0 + \mu\mathbf{u}_1 + ...)\end{aligned} \tag{B.23}$$

$\mathbf{h}_0$, $\mathbf{h}_1$ are obtained by equating the powers of μ in the manifold condition

$$\mathbf{h}_0 = (\mathbf{J}^{-1} + \mathbf{M}_l^{-1})^{-1}(\mathbf{M}_l^{-1}\mathbf{F}_r + \mathbf{J}^{-1}\mathbf{u}_0) \tag{B.24}$$

$$\mathbf{h}_1 = (\mathbf{J}^{-1} + \mathbf{M}_l^{-1})^{-1}(-\ddot{\mathbf{h}}_0 + \mathbf{J}^{-1}\mathbf{u}_1) \tag{B.25}$$

The reduced order flexible model becomes

$$\begin{aligned}\ddot{\theta}_l \; = \; & -\mathbf{M}_l^{-1}\mathbf{F}_r + \mathbf{M}_l^{-1}(\mathbf{J}^{-1} + \mathbf{M}_l^{-1})^{-1}\mathbf{M}_l^{-1}\mathbf{F}_r \\ & +\mathbf{M}_l^{-1}(\mathbf{J}^{-1} + \mathbf{M}_l^{-1})^{-1}\mathbf{J}^{-1}\mathbf{u}_0 - \mu\mathbf{M}_l^{-1}(\mathbf{J}^{-1} + \mathbf{M}_l^{-1})^{-1}\ddot{\mathbf{h}}_0 \\ & +\mu\mathbf{M}_l^{-1}(\mathbf{J}^{-1} + \mathbf{M}_l^{-1})^{-1}\mathbf{J}^{-1}\mathbf{u}_1 + \mathcal{O}(\mu^2) \\ \\ = \; & -(\mathbf{M}_l + \mathbf{J})^{-1}\mathbf{F}_r + (\mathbf{M}_l + \mathbf{J})^{-1}\mathbf{u}_0 - \mu(\mathbf{M}_l + \mathbf{J})^{-1}\mathbf{J}\ddot{\mathbf{h}}_0 \\ & +\mu(\mathbf{M}_l + \mathbf{J})^{-1}\mathbf{u}_1 + \mathcal{O}(\mu^2)\end{aligned} \tag{B.26}$$

The reduced order flexible model reduces to the rigid model as the perturbation parameter μ tends to zero.

The corrective control $\mathbf{u}_1$ is simply designed to annihilate the μ term in the reduced–order flexible model:

$$\mathbf{u}_1 = \mathbf{J}\ \ddot{\mathbf{h}}_0$$

To implement $\mathbf{u}_1$, we need to express $\ddot{\mathbf{h}}_0$ in terms of the available signals θ_l and $\dot{\theta}_l$. Differentiating B.24 gives

$$\dot{\mathbf{h}}_0 \; = (\mathbf{J}^{-1} + \mathbf{M}_l^{-1})^{-1}[\mathbf{M}_l^{-1}\dot{\mathbf{F}}_r + \mathbf{J}^{-1}(\mathbf{M}_l + \mathbf{J})\dot{v} + \mathbf{J}^{-1}\dot{\mathbf{F}}_r]$$

$$\begin{aligned}\ddot{\mathbf{h}}_0 \; & = (\mathbf{J}^{-1} + \mathbf{M}_l^{-1})^{-1}\left[(\mathbf{M}_l^{-1} + \mathbf{J}^{-1})\ddot{\mathbf{F}}_r + \mathbf{J}^{-1}(\mathbf{M}_l + \mathbf{J})\ddot{v}\right] \\ & = \ddot{\mathbf{F}}_r + \mathbf{M}_l\ \ddot{\mathbf{v}}\end{aligned}$$

The total control torque is then given by

$$\begin{aligned}\mathbf{u} \; & = \mathbf{u}_0 + \mu\ \mathbf{u}_1 \\ & = (\mathbf{M}_l + \mathbf{J})\mathbf{v} + \mathbf{F}_r(\theta_l, \dot{\theta}_l) + \mathbf{K}_j^{-1}\mathbf{J}(\ddot{\mathbf{F}}_r + \mathbf{M}_l\ddot{\mathbf{v}})\end{aligned} \tag{B.27}$$

with

$$\mathbf{v} = \ddot{\theta}_{ld} + \mathbf{K}_v(\dot{\theta}_{ld} - \dot{\theta}_l) + \mathbf{K}_p(\theta_{ld} - \theta_l)$$

If the singular perturbation solution is expressed as a feedforward compensation scheme, the control torque becomes

$$\mathbf{T} = \mathbf{T}_{ff} + \mathbf{T}_{fb} \tag{B.28}$$

with

$$\begin{aligned} \mathbf{T}_{ff} &= (\mathbf{M}_l + \mathbf{J})\,\ddot{\boldsymbol{\theta}}_{ld} + \mathbf{F}_r(\boldsymbol{\theta}_{ld}, \dot{\boldsymbol{\theta}}_{ld}) + \mathbf{K}_j^{-1}\mathbf{J}\,\ddot{\mathbf{F}}_r(\boldsymbol{\theta}_{ld}, \dot{\boldsymbol{\theta}}_{ld}) \\ &\quad + \mathbf{J}\mathbf{K}_j^{-1}\mathbf{M}_l\boldsymbol{\theta}_{ld}^{IV} \\ &= -\mathbf{K}_j\,\mathbf{q}_{jd} + \mathbf{J}\,\ddot{\boldsymbol{\theta}}_{md} \end{aligned} \tag{B.29}$$

and

$$\begin{aligned} \mathbf{T}_{fb} &= (\mathbf{M}_l + \mathbf{J})\,[\,\mathbf{K}_v(\dot{\boldsymbol{\theta}}_{ld} - \dot{\boldsymbol{\theta}}_l) + \mathbf{K}_p(\boldsymbol{\theta}_{ld} - \boldsymbol{\theta}_l)\,] \\ &\quad + \mathbf{J}\mathbf{K}_j^{-1}\mathbf{M}_l\,[\,\mathbf{K}_v(\boldsymbol{\theta}_{ld}^{III} - \boldsymbol{\theta}_l^{III}) + \mathbf{K}_p(\ddot{\boldsymbol{\theta}}_{ld} - \ddot{\boldsymbol{\theta}}_l)\,] \end{aligned} \tag{B.30}$$

B.5 Comparison of the three solutions

If the feedforward compensation scheme is applied, the same feedforward compensation torque is obtained for the three methods. The methods differ in the choice of feedback variables. For the computed torque and singular perturbation solution the feedback variables are the link position, velocity, acceleration and jerk: $\boldsymbol{\theta}_l$, $\dot{\boldsymbol{\theta}}_l$, $\ddot{\boldsymbol{\theta}}_l$ and $\boldsymbol{\theta}_l^{III}$. For the generalized nonlinear decoupling solution the feedback variables are the motor position and velocity and the flexible deformation and velocity: $\boldsymbol{\theta}_m$, $\dot{\boldsymbol{\theta}}_m$, $\mathbf{q}_j = \boldsymbol{\theta}_m - \boldsymbol{\theta}_l$ and $\dot{\mathbf{q}}_j = \dot{\boldsymbol{\theta}}_m - \dot{\boldsymbol{\theta}}_l$.
If it is supposed that exact compensation is achieved, the computed torque method and singular perturbation method do not need linearization for feedback gain calculation. The generalized nonlinear decoupling method needs linearization of the partially linearized model for feedback gain calculation.

Appendix C

Trajectory generation and inverse kinematics

The objective of trajectory planning is to connect a number of prespecified points by a smooth curve.

The smoother the curve, the better the performance with respect to tracking accuracy and excitation of structural dynamics [75]. The smoothness of the trajectory is determined by the order of the derivative which is the first to possess a discontinuity. As a fundamental requirement for control of rigid manipulators, it is generally accepted that the acceleration must be jump free. Or in other words that the rate of change of the acceleration (and consequently of the torques applied) may not become infinite. Link accelerations can be changed instantly by changing the motor torques, however, stepped acceleration commands have very high frequency content and they tend to excite vibrations. Smooth acceleration commands prevents high dynamic loads on the mechanical structure, vibrations, noise and wear.

Control of flexible manipulators requires a jump free rate of change of jerk. This avoids discontinuities in the feedforward torques (computed with the computed torque or generalized nonlinear decoupling control method). As for rigid manipulators discontinuities in the torques need to be avoided, to avoid unnecessary excitation of resonance frequencies.

Section C.1 generates a trajectory, satisfying the requirement of jump free acceleration, by means of an imposed parabolic acceleration profile. Section C.2 generates a trajectory by means of a polynomial

spline fit. A fifth order trajectory satisfies the requirement of jump free acceleration. A ninth order trajectory satisfies the requirement of jump free rate of change of jerk.

Various manufacturing tasks such as conveyor–belt tracking, arc welding, spray painting, assembly, and many others, need control of manipulator motion in operational space. Therefore first a cartesian trajectory for the tip (or the hand) of the manipulator is generated. The control of manipulators at the joint level, requires the transformation of the cartesian trajectory into the corresponding joint trajectories, the so called inverse–kinematic problem. Section C.3 derives the trajectories for the joint angles of a planar two–link manipulator for a straight line movement of the tip.

C.1 Trajectory generation in the operational space by means of an imposed parabolic acceleration profile

Trajectory generation by means of an imposed parabolic acceleration profile is described in [17].
Consider a general one dimensional position variable x, with a total displacement x_{tot} and with derivatives v_{max} (desired maximum velocity) and a_{max} (desired maximum acceleration). A total time duration T can be specified.
For describing the acceleration profile a linear jerk profile is proposed. The jerk is the derivative of the acceleration. By successive integrations of this profile the acceleration profile, the velocity profile and the displacement profile are found (figure C.1). These profiles are described using the following parameters:

1. α = the gradient of the jerk profile
2. T_1= the time duration for reaching the maximum value of the jerk
3. T_2= the duration of the motion with constant velocity.

The relationship between these profile parameters and the specified total displacement, maximum velocity and maximum acceleration is given in [17]. Also is explained how to proceed if a total time duration T has been specified.

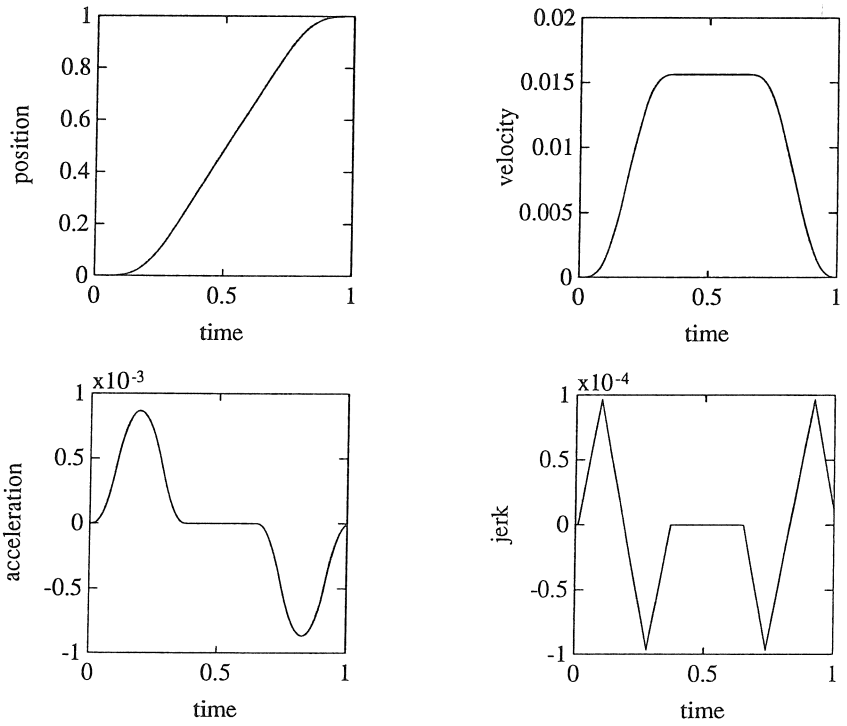

Figure C.1: Trajectory generation by means of an imposed parabolic acceleration profile

C.2 Trajectory generation in the operational space by means of a polynomial spline fit

A polynomial spline fit gives a smooth trajectory between initial and final point.
The basic formula for a nth order polynomial is

$$p(t) = a_n\, t^n + a_{n-1}\, t^{n-1} + \ \ldots\ + a_1\, t + a_0 \qquad \text{(C.1)}$$

with, $0 \leq t \leq 1$, such that $0 \leq p(t) \leq 1$. The order n of the polynomial must be odd. The following boundary conditions determine the coefficients in this basic formula:

$$p(t)|_{t=0} = 0,$$

$$p(t)|_{t=1} = 1,$$

$$\frac{d^i p(t)}{dt^i}|_{t=0} = 0,$$

$$\frac{d^i p(t)}{dt^i}|_{t=1} = 0,$$

for i=1, 2, ..., $\frac{n-1}{2}$.

Fifth order polynomials allow matching the positions, velocities, and accelerations at both ends of each path segment. They do not make discontinuous changes in acceleration; however they do make step changes in jerk, the third derivative. The basic formula for the fifth order polynomial is

$$p(t) = 6\, t^5 - 15\, t^4 + 10\, t^3 \qquad \text{(C.2)}$$

A ninth order polynomial does not make discontinuous changes in rate of change of jerk, the fourth derivative. The basic formula for the ninth order polynomial is

$$p(t) = 70\, t^9 - 315\, t^8 + 540\, t^7 - 420\, t^6 + 126\, t^5 \qquad \text{(C.3)}$$

Figure C.2 compares a set of time histories for position, velocity, acceleration, and jerk of a fifth–order polynomial trajectory and a ninth–order polynomial trajectory. For the same trajectory duration, using higher order polynomials will require higher maximum accelerations. Because saturation of motors limits the maximum achievable

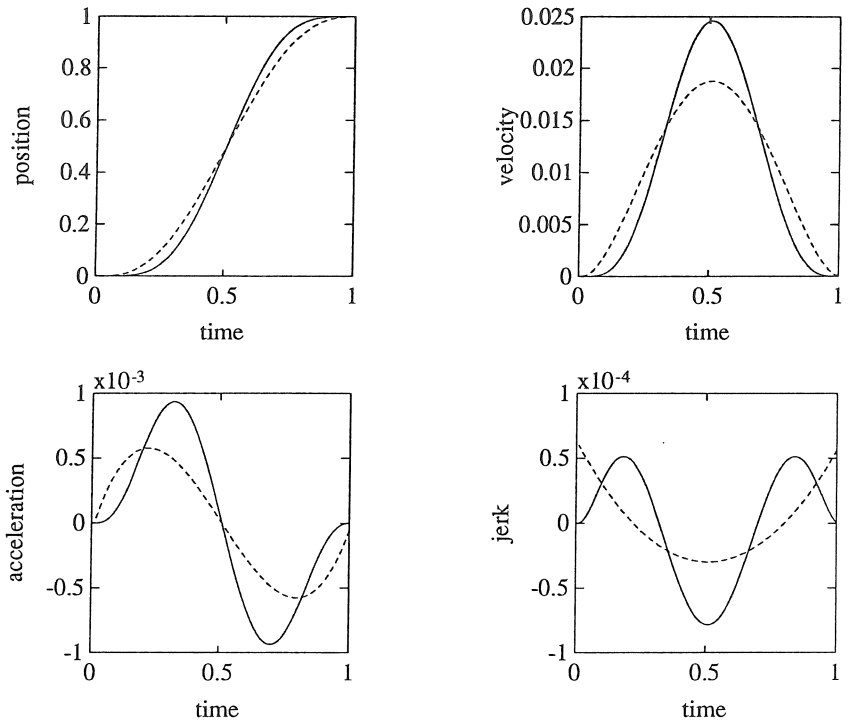

Figure C.2: Fifth–order (dashed line) and ninth–order (solid line) polynomial trajectory

acceleration, it will be necessary to increase the trajectory duration. The choice of the order of the polynomial is thus a compromise between smoothness (and thus improved performance with respect to tracking accuracy and excitation of structural dynamics) and the speed of the robot.

C.3 Example: straight–line movement by a two–link manipulator

This section derives the trajectories for the joint angles of a planar two–link manipulator (figure C.3) for a straight line movement of the tip.

A straight line movement of the tip is given by $y - y_0 = (x - x_0)\frac{y_1 - y_0}{x_1 - x_0}$

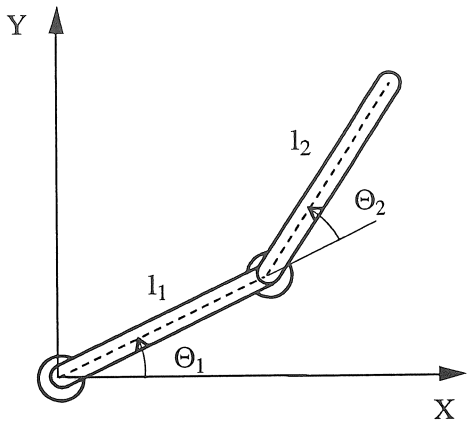

Figure C.3: A planar two–link manipulator

for the beginning and end positions of the tip (x_0, y_0) and (x_1, y_1), respectively.
The control of manipulators at the joint level, requires to transform from the position, velocity, and acceleration of the tip to the position, velocity, and acceleration for each joint angle. These inverse kinematic equations are presented below for a planar two–link manipulator [38]:

$$\begin{aligned} cos\theta_2 &= \frac{x^2+y^2-l_1^2-l_2^2}{2l_1l_2} \\ \theta_1 &= tan^{-1}\left(\frac{y}{x}\right) - tan^{-1}\left(\frac{l_2 sin\theta_2}{l_1+l_2 cos\theta_2}\right) \end{aligned} \tag{C.4}$$

$$\begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_1+\dot{\theta}_2 \end{bmatrix} = \frac{1}{l_1 l_2 sin\theta_2} \begin{bmatrix} l_2 cos(\theta_1+\theta_2) & l_2 sin(\theta_1+\theta_2) \\ -l_1 cos\theta_1 & -l_1 sin\theta_1 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} \tag{C.5}$$

$$\begin{aligned} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_1+\ddot{\theta}_2 \end{bmatrix} &= \frac{1}{l_1 l_2 sin\theta_2} \begin{bmatrix} l_2 cos(\theta_1+\theta_2) & l_2 sin(\theta_1+\theta_2) \\ -l_1 cos\theta_1 & -l_1 sin\theta_1 \end{bmatrix} \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} \\ &+ \frac{1}{l_1 l_2 sin\theta_2} \begin{bmatrix} l_1\, l_2 cos\theta_2 & l_2^2 \\ -l_1^2 & -l_1 l_2 cos\theta_2 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1^2 \\ (\,\dot{\theta}_1+\dot{\theta}_2\,)^2 \end{bmatrix} \end{aligned} \tag{C.6}$$

Bibliography

[1] M. Adams, R. Buckincx, *Een efficiënte en betrouwbare berekeningsmethode voor de totale kleinste kwadratenbenadering.* Eng. Thesis, Dept. of Electrical Engineering, K.U.Leuven, Leuven, Belgium, 1986.

[2] M. Adams, J. De Schutter, *Identification and control of a single degree of freedom positioning system.* Proc. 13th Int. seminar on modal analysis, K.U.Leuven, Leuven, Belgium, 1988.

[3] M. Adams, J. De Schutter, *Experiments on nonlinear control of flexible robots.* Proc. 15th Int. seminar on modal analysis, K.U.Leuven, Leuven, Belgium, 1990.

[4] M. Adams, J. Swevers, D. Torfs, J. De Schutter, H. Van Brussel, *Implementation of control algorithms for flexible joint robots on a KUKA IR 161/60 industrial robot: experimental results.* Proc. 2th Int. Symposium on Experimental Robotics, Toulouse, France, 1991.

[5] B. Amstrong, O. Khatib, J. Burdicck, *The explicit dynamic model and inertial parameters of the PUMA 560 Arm.* Proc. IEEE Conf. on Robotics and Automation, San Francisco, California, 1986.

[6] H. Asada, Z.D. Ma, H. Tokumara, *Inverse dynamics of flexible robot arms: modeling and computation for trajectory control.* ASME Journal of Dynamic Systems, Measurement and Control, Vol. 112, pp. 177–185, June 1990.

[7] K.J. Åström, B. Wittenmark, *Computer controlled systems.* Englewood Cliffs, N.J.: Prentice Hall, 1984.

[8] M. Benati, A. Morro, *Dynamics of chain of flexible links.* ASME Journal of Dynamic Systems, Measurement, and Control, Vol. 110, pp. 410–415, 1988.

[9] S.A. Bortoff, M.W. Spong, *Feedback linearization of flexible joint robot manipulators.* Proc. 26th IEEE Conf. on Decision and Control, Los Angeles, pp. 1357–1362, 1987.

[10] P. Buysse, M. Christiaens, *De optimalisatie van de kontrole van een flexibele robotarm.* Eng. Thesis, Dept. of Mechanical Engineering, K.U. Leuven, Leuven, Belgium, 1989.

[11] R.H. Cannon, E. Schmitz, *Initial experiments on the Endpoint control of a flexible one-link robot.* International Journal of Robotics Research, Vol. 3, No. 3, pp. 62–75, 1984.

[12] C. Canudas De Wit, O. Lys, *Robust control and parameter estimation of robots with flexible joints.* Proc. IEEE Conf. on Robotics and Automation, Philadelphia, pp. 324–329, 1988.

[13] C. Canudas De Wit, P. Noel, A. Aubin, B. Brogliato, P. Drevet, *Adaptive friction compensation in robot manipulators: low–velocities.* Proc. IEEE Conf. on Robotics and Automation, pp. 1352–1357, 1989.

[14] S. Cetinkunt, W.J. Book, *Symbolic modeling and dynamic simulation of robotic manipulators with compliant links and joints.* Robotics and Computer–integrated Manufacturing, Vol. 5, No. 4, pp. 301–310, 1989.

[15] A. De Luca, A. Isidori, F. Nicolo, *An application of nonlinear model matching to the dynamic control of robot arm with elastic joints via nonlinear dynamic feedback.* Proc. 24th IEEE Conf. on Desicion and Control, Ft. Lauderdale, Florida, pp. 1671–1679, 1985.

[16] A. De Luca, *Dynamic control of robots with joint elasticity.* Proc. IEEE Conf. on Robotics and Automation, Philadelphia, pp. 152–158, 1988.

[17] F. Demeester, *Selection of applicable off-line computation algorithms for trajectory control.* Technical report, task 3154, Esprit project 1561:Sacody, K.U. Leuven, Leuven, Belgium, 1987.

[18] J. De Schutter, L. Vastmans, J. Simons, *Niet–gekoppelde kontrole.* Zomercursus; mathematica, programmatie en controle van industriele robots, Dept. of Mechanical Engineering, K.U.Leuven, Leuven, Belgium, pp. 108-123, 1983.

[19] J. De Schutter, J. Simons, *Gekoppelde kontrole.* Zomercursus; mathematica, programmatie en controle van industriele robots, Dept. of Mechanical Engineering, K.U.Leuven, Leuven, Belgium, pp. 124-139, 1983.

[20] J. De Schutter, H. Van Brussel, M. Adams, A. Froment, J.L. Faillot, *Control of flexible robots using generalized nonlinear decoupling.* Proc. Symposium on Robot Control, Karlsruhe, pp. 98.1–98.6, 1988.

[21] J. De Schutter, *An introduction to State Space Control and its Application to Motion Control.* Computer controlled motion and robotics course, K.U.Leuven, Production engineering, machine design and automation, Leuven, Belgium, pp. 105–134, 1991.

[22] S. Dubowsky, *On the adaptive control of robotic manipulators.* Proc. Joint Autom. Control Conf., Charlotteville, USA, 1981.

[23] M. Duque, M. Samaan, *Partial state LQ and GPC adaptive control: An experimental evaluation.* Proc. 8th Conf. on Analysis and Optimization of Systems, Springer-Verlag, 1988.

[24] P. Eykhoff, *System Identification.* Wiley, New York, 1974.

[25] J.L. Faillot, *Examine state of the art on control of flexible structures.* Technical report, task 3121, Esprit project 1561:Sacody, Bertin, France, 1987.

[26] M.G. Forrest-Barlach, S.C. Babcock, *Inverse dynamic position control of a compliant manipulator.* IEEE Journal of Robotics and Automation, Vol. RA-3, No. 1, pp. 75–83, 1987.

[27] G.F. Franklin, J.D. Powell, *Digital control of dynamic systems.* Addison-Wesley, 1980.

[28] E. FREUND, *Fast nonlinear control with arbitrary pole-placement for industrial robots and manipulators.* International Journal of Robotics Research, 1, pp. 65–78, 1982.

[29] A. FROMENT, J.L. FALLOT, M. ADAMS, J. DE SCHUTTER, *Selection of applicable off-line computation algorithms for tracking control. Preliminary computation of tracking control laws. Numerical computation and verification on example of limited flexibility, then on generic model.* Technical report, task 3151, 3152, 3153, Esprit project 1561:Sacody, Bertin, France and K.U. Leuven, Leuven, Belgium, 1988.

[30] B. GEBLER, *Modellbildung, Steuerung und Regelung fur elastische Industrieroboter.* Ph.D. Thesis, Technische Universität München, VDI Verlag, Reihe 11: schwingungstechnik nr. 98, 1987.

[31] B. GEBLER, *Feedforward control strategy for an industrial robot with elastic links and joints.* Proc. IEEE Conf. on Robotics and Automation, Raleigh, pp. 923–928, 1987.

[32] A.A. GOLDENBERG, A. KELLY, *A kinematics and dynamics simulation of the IBM 6765 robot using AML.* Journal of Robotic Systems 2(4), pp. 353–371, 1985.

[33] G. GOLUB, C. VAN LOAN, *Matrix Computations.* North Oxford Academic Publ. Co., John Hopkins University Press, 1983.

[34] G.C. GOODWIN, K.S. SIN, *Adaptive filtering predicion and control.* Englewood Cliffs, N.J.: Prentice Hall, 1984.

[35] V. HELD, C. MARON, *Estimation of friction characteristics, inertial and coupling coefficients in robotic joints based on current and speed measurements.* Proc. Symposium on Robot Control, Karlsruhe, 1988.

[36] M.G. HOLLARS, *Experimental implementation of a nonlinear estimator in the control of flexible joint manipulators.* Submitted to the 1989 IFAC Conf. Tsukuba, Japan, 1989.

[37] M.G. HOLLARS, *Experiments in end-point control of manipulators with elastic drives.* Ph.D. Thesis, Stanford University, Stanford, California, Department of Aeronautics and Astronautics, 1988.

[38] J.M. HOLLERBACH, *Dynamic scaling of manipulator trajectories.* ASME Journal of Dynamic Systems, Measurement and Control, Vol. 106, pp. 102–106, 1984.

[39] T.C. HSIA, *Adaptive control of robot manipulators – a review.* Proc. IEEE Conf. on Robotics and Automation, San Francisco, pp. 183–189, 1986.

[40] C. JANKOWSKI, *Modelling of a one–flexible link manipulator.* Technical report, Esprit project 1561:Sacody, K.U.Leuven, Leuven, Belgium, 1988.

[41] H. KANOH, S. TZAFESTAS, H.G. LEE, J. KALAT, *Modelling and control of flexible robot arms.* Proc. IEEE Conf. on Decision and Control, pp. 1866–1870, 1986.

[42] M. KHORRAMI, Ü. ÖZGÜNER, *Perturbation methods in control of flexible link manipulators.* Proc. IEEE Conf. on Robotics and Automation, Philadelphia, pp. 310–315, 1988.

[43] P.K. KHOSLA, T. KANADA, *Parameter identification of robot dynamics.* Proc. 24th IEEE Conf. on Decision and Control, Ft. Lauderdale, Florida, pp. 1754–1760, 1985.

[44] P.K. KHOSLA, T. KANADA, *Experimental evaluation of nonlinear feedback and feedforward control schemes for manipulators.* Technical report, Department of Electrical and Computer Engineering, The Robotics Institute, Carnegie Mellon Univesity, Pittsburgh, Pennsylvania 15213, 1987.

[45] A.J. KOIVO, *Self–tuning manipulator control in cartesian base coordinate system.* ASME Journal of Dynamic Systems, Measurement and Control, Vol. 107, pp. 316–323, 1985.

[46] V.V. KOROLOV, Y.H. CHEN, *Robust control of a flexible manipulator arm.* Proc. IEEE Conf. on Robotics and Automation, Philadelphia, pp. 159–164, 1988.

[47] T. KUBO, G. ANWAR, M. TOMIZUKA, *Application of nonlinear friction compensation to robot arm control.* Proc. IEEE Conf. on Robotics and Automation, San Francisco, California, pp. 722–726, 1986.

[48] H.B. KUNTZE, *Position control of industrial robots – impacts, concepts and results.* Proc. Symposium on Robot Control, Karlsruhe, 1988.

[49] L. LJUNG, T. SODERSTROM, *Theory and practice of recursive identification.* Cambridge: MIT Press, 1983.

[50] L. LJUNG, *System identification: theory for the user.* Englewood Cliffs, N.J.: Prentice Hall, 1987.

[51] LMS INTERNATIONAL *KUKA IR 161 modal analysis results.* Technical report, Esprit project 1561:Sacody, LMS International, Belgium, 1990.

[52] R. MARINO, S. NICOSIA *Singular perturbation techniques in the adaptive control of elastic robots.* Proc. Symposium on Robot control, Barcelona, 1985.

[53] M. MERTENS, H. VAN DER AUWERAER, *The complex stiffness method to detect and identify nonlinear dynamic behaviour of SDOF systems.* Mechanical Systems and Signal Processing, Vol. 3, No. 1, pp. 37–54, 1989.

[54] C.M. OAKLY, R.H. CANNON, *Theory and experiments in selecting mode shapes for two–link flexible manipultors.* Stanford University Aerospace Robotics Laboratory, Stanford, California 94305, USA, 1990.

[55] F. PFEIFFER, B. GEBLER, *A Multistage approach to the dynamics and Control of elastic robots.* Proc. IEEE Conf. on Robotics and Automation, Philadelphia, pp. 2–8, 1988.

[56] A. PREUMONT, *Spillover alleviation for nonlinear active control of vibration.* J. Guidance, Vol. 11, No. 2, pp. 124–130, 1987.

[57] *Pro–Matlab, Vax/VMS version 3.5f., VMS.* Mathworks Inc. MA, USA, December 1989.

[58] S. RAMAKRISHNAN, *Experimental identification and control of the tip position of a flexible, single link manipulator.* Ph.D. Thesis, University of California, Berkeley, Department of Mechanical Engineering, 1985.

[59] K. ROBBROECKX, K. DE MEESTER, *Niet-lineaire identificatie en controle van een robot met twee gelederen en een flexibel gewricht.* Eng. Thesis, Dept. of Mechanical Engineering, K.U. Leuven, Leuven, Belgium, 1990.

[60] M.D. ROVNER, *Experiments toward on-line identification and control of a very flexible one-link manipulator.* International Journal of Robotics Research, Vol. 6, No. 3, pp. 3–19, 1987.

[61] Y. SAKAWA, F. MATSUNO, S. FUKUSHIMA, *Modeling and feedback control of a flexible arm.* Journal of Robotic Systems, Vol. 2, No. 4, pp. 453–472, 1985.

[62] B. SICILIANO, W.J. BOOK, *A singular perturbation approach to control of lightweight flexible manipulators.* International Journal of Robotics Research, Vol. 7, No. 4, pp. 79–90, 1988.

[63] M. SIDMAN, *Adaptive control of a flexible structure.* Ph.D. Thesis, Stanford University, Department of Electrical Engineering, 1986.

[64] S.N. SINGH, A.A. SCHY, *Robust torque control of an elastic robotic arm based on invertibility and feedback stabilization.* Proc. 24th IEEE Conf. on Decision and Control, Ft. Lauderdale, Florida, 1985.

[65] M.W. SPONG, K. KHOROSANI, P.V. KOKOTOVIC, *An integral manifold approach to the feedback control of flexible joints robots.* IEEE Journal of Robotics and Automation, Vol. 3, No. 4, pp. 291–300, 1987.

[66] J. SWEVERS, M. ADAMS, J. DE SCHUTTER, H. VAN BRUSSEL, H. THIELEMANS, *Limitations of linear identification and control techniques for flexible Robots with nonlinear joint friction.* First Int. Symposium on Experimental Robotics, Montreal, Canada, Springer–Verlag, 1990.

[67] J. SWEVERS, *An introduction to system identification and its application in practice.* Computer controlled motion and robotics course, K.U.Leuven, Production engineering, machine design and automation, Leuven, Belgium, pp. 47–76, 1991.

[68] J. SWEVERS, D. TORFS, M. ADAMS, J. DE SCHUTTER, H. VAN BRUSSEL, *Comparison of control algorithms for flexible joint robots implemented on a KUKA IR 161/60 industrial robot.* Proc. 5th Int. Conf. on Advanced Robotics, Pisa, Italy, 1991.

[69] J. SWEVERS, *Linear identification and control of flexible robots.* Ph.D. Thesis, Dept. of Mechanical Engineering, K.U.Leuven, Leuven, Belgium, 1992.

[70] M. TAHARA, S. CHONAN, *Closed–loop displacement control of a one–link flexible arm with a tip mass.* JSME International Journal, Series III, Vol. 31, No. 2, pp. 409–415, 1988.

[71] M. TOMIZUKA, *Zero phase error tracking algorithm for digital control.* Journal of Dynamic Systems, Measurements, and Control, Vol. 109, pp. 65–68, 1987.

[72] D. TORFS, J. SWEVERS, J. DE SCHUTTER, *Quasi–perfect tracking control of non–minimal phase systems* Proc. IEEE Conf. on Decision and Control, Brighton, 1991.

[73] A. M. TRAAS, *Precise control of flexible manipulators.* Journal A: Vol. 29, No. 2, pp. 3–9, 1988.

[74] I. TROCH, *Control concepts and algorithms for flexible robots – an expository survey.* Proc. Symposium on Robot Control, Karlsruhe, pp. 2.1–2.6, 1988.

[75] L. VAN AKEN, *Robot motions in free space: task specification and rajectory planning.* Ph.D. Thesis, Dept. of Mechanical Engineering, K.U.Leuven, Leuven, Belgium, 1987.

[76] J. VAN DEN BOSSCHE, *RODYM: a new approach to robot metrology.* Technical report, Dept. of Mechanical Engineering, K.U.Leuven, Leuven, Belgium, 1989.

[77] H. VAN DER AUWERAER, *Development and evaluation of advanced measurement methods for experimental modal analysis.* Ph.D. Thesis, Dept. of Mechanical Engineering, K.U.Leuven, Leuven, Belgium, 1987.

[78] S. VAN HUFFEL, J .VANDEWALLE, *The generalized total linear least squares problem.* Siam Journal on Matrix Analysis and Applications, Vol. 10, No. 3, pp. 294–315, July 1989.

[79] R. VOSSOUGHI, M. DONATH, *Robot finger stiffness control in the presence of mechanical nonlinearities.* ASME Journal of Dynamic Systems, Measurement, and Control, Vol. 110, pp. 236–245, 1988.

[80] W.J. WANG, S.S. LU, C.F. HSU, *Experiments on the position control of a one–link flexible robot arm.* IEEE transactions on Robotics and Automation, Vol.5, No. 3, pp. 373–377, 1989.

[81] K. WYCKAERT, *Development and evaluation of detection and identification schemes for the nonlinear dynamical behaviour of mechanical structures.* Ph.D. Thesis, Dept. of Mechanical Engineering, K.U.Leuven, Leuven, Belgium, 1992.

[82] S. YANG, M. TOMIZUKA, *Adaptive pulse width control for precise positioning under the influence of stiction and coulomb friction.* ASME Journal of Dynamic Systems, Measurement, and Control, Vol. 110, pp. 221-227, 1988.

[83] H. YAMAURA, K. ONO, *Vibrationless starting and stopping control for a flexible arm.* JSME International Journal, Series III, Vol. 32, No. 3, 1989.